



# **Web based Bookstore Management System for Wisdom**

**Name** : Shamila Indunil Koralage  
(S.I. Koralage)

**BIT Registration No** : R050916

**Index No** : 0509167

**Name of Supervisor** : Mohamed Shafraz Razick

**December 2017**

BIT

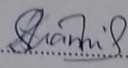


**This dissertation is submitted in partial fulfillment of the requirement of the  
Degree of Bachelor of Information Technology (external) of the  
University of Colombo School of Computing**

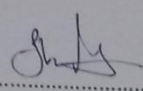
# DECLARATION

## Declaration

"I certify that this dissertation does not incorporate, without acknowledgement, any materials previously submitted for a degree or diploma in any university and to the best of my knowledge and belief, it does not contain any material previously published or written by another person or myself except where due reference is made in text. I also hereby give consent for my dissertation, if accepted, to be made available for photocopying and for interlibrary loans, and for the title and abstract to be made available to outside organizations."

Signature of Candidate:  Date: 05/11/17

Name of Candidate: **S.I. Koralage**

Countersigned by:  
Signature of Supervisor/Advisor:  Date: 05/11/17

Name of Supervisor / Advisor: **Mr. Mohamed Shafraz Razick**

# ABSTRACT

Education has been the most critical factor in the present knowledge era, not only in the world but also in Sri Lanka. Every parent tries to provide the best education for their children, hence there has been a rapid demand for the popular national schools and international schools. Therefore, the demand for international school syllabus textbooks has been also increasing, in addition to the demand for local school additional texts. Because of that, there is a big opportunity in selling international texts and local syllabus related texts for the booksellers in the country.

WISDOM BOOKSHOP (WB) is one of the well-recognized, ongoing bookshops and also WB was established 20 years ago in Wattala area. Presently plays a vital role in International textbooks for Local, London – O/L, A/L sections in bookshop segment. It only handles retail customer's book Sales at begins; later caters through the books from Dealers (suppliers). Then Wisdom Bookshop combines with several Dealers with Wholesale and Retail sales with Agents in bookshop Sector. At present they have their own website and use IT facilities to some extent in their daily internal business process but based on poor semi-automated Computer System. This makes the business inefficiently, Data inaccuracy and unavailability so it causing its future progress although they are able to reach their customers via the web. WEB BASED BOOKSTORE MANAGEMENT (WBBM) is providing a solution for the key problems in the internal business process. This solution is built in a way to cover the overall processes of the bookstore such as Purchase, Sales, Stock controlling and Report Generation.

The System has tailored Model View Controller (MVC) Architecture and Object Oriented (OOAD) techniques with UML were used to analyze and design the system. The development process was preliminary based on the Rational Unified Process (RUP). HTML5 was used as the core markup language while CSS3 for styling. PHP use for server-side scripting, JavaScript, jQuery and AJAX for client-side scripting. MySQL was used as the DBMS of the system. WBBM would improve the efficiency and productivity of the daily functions of the business, and also would strengthen the suppliers – business - customers' relationship which leads to adding competitive advantage to the business in for future progress. The online distributed nature of the system will be very helpful to expand their business in the future as well.

# ACKNOWLEDGEMENT

I take this Opportunity to say my Thanks to everyone who helped me in various ways and guided to me to complete this Project successfully.

My First thank must go to Project Supervisor Mr Mohamed Shafraz Razick, Consultant of Virtusa and Lecturer of OpenArc (Pvt) Ltd, Colombo 05, he gave the guidance and also a Lecturer of OpenArc Mr Ishantha Dhanushka. It was a great help for all of us who studied in that Institute to get a clear idea about this Project work towards the success.

Mr Alloy M. Robert, the owner of Wisdom Bookshop – Wattala, cater me to their knowledge and necessary information of their Bookshop for my domain improvement for regarding Project to bring a reality to their Bookshop. My thanks also go to the Wisdom Bookshop Staff, who supported me about the processes and functionalities of the Bookshop transactions and assist in the course of Development of the Project.

I wish to state my recognition to the BIT Coordinator of University of Colombo School of Computing (UCSC) and Project Examination Board of Bachelors of Information Technology (BIT) for giving me the prospect to apply my theory and practical knowledge to achieve through the BIT degree. The Individual Project is the most important work from all the three years of Studying the Bachelor of Information Technology.

Finally, also grateful to My Parents and relation for supporting and encouraging me to complete this Project Successfully. In this hard process, there are so many friends who helped me and guided me towards the success.

# TABLE OF CONTENTS

<b>DECLARATION</b> .....	<b>ii</b>
<b>ABSTRACT</b> .....	<b>iii</b>
<b>ACKNOWLEDGEMENT</b> .....	<b>iv</b>
<b>TABLE OF CONTENT</b> .....	<b>v</b>
<b>LIST OF FIGURES</b> .....	<b>vii</b>
<b>LSIT OF TABLES</b> .....	<b>ix</b>
<b>LIST OF ACRONYMS</b> .....	<b>x</b>
<b>CHAPTER 1 - INTRODUCTION</b> .....	<b>1</b>
1.1 Introduction.....	1
1.2 The Client .....	1
1.3 Motivation.....	2
1.4 Objectives.....	2
1.5 Scope.....	3
1.6 Structure of the Dissertation .....	4
<b>CHAPTER 2 - ANALYSIS</b> .....	<b>5</b>
2.1 Introduction.....	5
2.2 Fact Gathering Techniques .....	5
2.3 The Existing system.....	6
2.3.1 Purchase Module .....	6
2.3.2 Sales Module .....	6
2.3.3 Stock Module .....	7
2.3.4 HR Module .....	7
2.4 Draw backs of Existing system.....	8
2.5 Functional Requirements .....	8
2.6 Non Functional Requirements .....	10
2.7 Literature Review .....	11
2.8 Development Methodology .....	12
<b>CHAPTER 3 - DESIGN</b> .....	<b>15</b>
3.1 Introduction.....	15
3.2 Development Strategy.....	15
3.3 Reason to choose the web based system .....	15
3.4 Design Techniques .....	16

3.5 Design of the WBBM.....	16
3.5.1 Purchase Module .....	17
3.5.2 Sales Module .....	19
3.5.3 Stock Module .....	20
3.5.4 Finance Module .....	22
3.5.5 Web Module .....	24
3.5.6 Report Module .....	25
3.5.7 Administration Module .....	26
3.6 Class Diagram of the WBBM.....	28
3.7 DB Design of the WBBM .....	28
3.8 Development Architecture.....	29
3.9 User interface Design of the WBBM.....	30
<b>CHAPTER 4 - IMPLEMENTATION .....</b>	<b>33</b>
4.1 Introduction .....	33
4.2 Implementation Environment .....	33
4.2.1 Development Tools .....	33
4.2.2 Development Technologies .....	34
4.2.3 Reused Modules .....	34
4.3 Implementation of CodeIgniter .....	35
4.4 Module Structure.....	36
4.5 Major Code Descriptions .....	36
4.5.1 Controller .....	36
4.5.2 Model .....	39
4.5.3 View .....	40
4.5.4 Library.....	42
<b>CHAPTER 5 - EVALUATION .....</b>	<b>44</b>
5.1 Introduction .....	44
5.2 System Test Plan.....	44
5.3 Testing Methods.....	44
5.4 Test Plan and Test Cases.....	45
5.4.1 Purchase Module .....	45
5.4.2 Sales Module.....	46
5.4.3 Stock Module.....	47

5.4.4 Finance Module.....	48
5.4.5 Web Module .....	49
5.4.5 Report Module.....	50
5.4.6 Admin Module.....	50
5.5 Test data and Test Results .....	51
5.6 Acceptance Test.....	51
<b>CHAPTER 6 - Conclusion .....</b>	<b>53</b>
6.1 Introduction .....	53
6.2 Critical Assessment of the Project.....	53
6.3 Future Improvements .....	53
6.4 Lessons Learnt .....	54
<b>REFERENCES.....</b>	<b>55</b>
<b>Appendix A   System Documentation .....</b>	<b>57</b>
<b>Appendix B   Design Documentation .....</b>	<b>60</b>
<b>Appendix C   User Documentation .....</b>	<b>72</b>
<b>Appendix D   Management Reports.....</b>	<b>81</b>
<b>Appendix E   Test Results .....</b>	<b>85</b>
<b>Appendix F   Code Listings .....</b>	<b>94</b>
<b>Appendix G   Client Certificate .....</b>	<b>99</b>
<b>Glossary.....</b>	<b>100</b>
<b>Index.....</b>	<b>101</b>

# List of Figures

Figure 2.1 Existing Bookshop System for wisdom.....	7
Figure 2.2 Two dimensional view of the RUP.....	13
Figure 3.1 Use case diagram for the Purchase Module.....	17
Figure 3.2 Use case diagram for the Sales Module .....	19
Figure 3.3 Use case diagram for the Stock Module.....	21
Figure 3.4 Use case diagram for the Financial Module.....	23
Figure 3.5 Use case diagram for the Web Module.....	24
Figure 3.6 Use case diagram for the Report Module.....	25
Figure 3.7 Use case diagram for the Administration Module.....	26
Figure 3.8 Sequence diagram for Add Users use case.....	27
Figure 3.9 Class diagram of the WBBM.....	28
Figure 3.10 Database Design of the WBBM.....	29
Figure 3.11 MVC Architecture.....	29
Figure 3.12 Login Interface for the WBBM.....	31
Figure 3.13 Administrator Home page.....	31
Figure 3.14 Data Grid interface of WBBM.....	32
Figure 3.15 Form Interfaces of WBBM.....	32
Figure 4.1 Implementation of the CI.....	35
Figure 4.2 Structure of the application.....	36
Figure B.1 Activity Diagram for Purchase Module.....	61
Figure B.2 Activity Diagram for Sales Module.....	62
Figure B.3 Sequence Diagram for Add Sales Module.....	63
Figure B.4 Sequence Diagram for Admin Module.....	68
Figure B.5 Deployment Diagram of the WBBM .....	68
Figure B.6 Classes of the WBBM.....	69
Figure B.7 Components of the WBBM.....	70
Figure B.8 Database structure for the WBBM.....	71
Figure C.1 Application Login.....	72
Figure C.2 Administrator Home page.....	72
Figure C.3 Password Reset page.....	73
Figure C.4 Link Bar .....	73



Figure C.5 Logout Area.....	73
Figure C.6 Table View of Data.....	74
Figure C.7 Data Grid.....	74
Figure C.8 Search Area.....	75
Figure C.9 Action Buttons.....	76
Figure C.10 Form Notifications.....	76
Figure C.11 Adding Personnel Information for new user .....	77
Figure C.12 Add New Purchase.....	78
Figure C.13 View of Purchase Page in WBBM.....	78
Figure C.14 Process of Purchase Requisition Notifications in WBBM.....	79
Figure C.15 Add Books to Stock in WBBM.....	79
Figure C.16 View of Books Stock in WBBM.....	80
Figure D.1 Monthly Sales reports details .....	79
Figure D.2 Monthly Stock reports per One Agent.....	79
Figure D.3 Annual Sales Report details of WBBM .....	80
Figure D.4 Annual Purchase Report details of WBBM .....	81
Figure D.5 Stock Category .....	82
Figure D.6 Purchase Order of WBBM.....	82
Figure D.7 Purchase GRN of WBBM .....	83
Figure D.8 Purchase Order Return of WBBM .....	83
Figure D.9 Sales Order of WBBM .....	83
Figure D.10 Stock, Sales, Purchase, Income, Expenditure Reports .....	84
Figure D.11 Summary of Stock per a day .....	84
Figure D.12 Payment Invoice .....	84
Figure E.1 User Acceptance Test Result Sheet.....	92
Figure E.2 User Evaluation Feedback Chart in WB.....	92

# List of Tables

Table 2.1 Literature Review.....	12
Table 3.1 Use case narrative for the Manage Purchase Orders use case.....	18
Table 3.2 Use case narrative for the Add Purchase Orders use case.....	18
Table 3.3 Use case narrative for Manage Sales use case.....	20
Table 3.4 Use case narrative for Search Book use case.....	20
Table 3.5 Use case narrative for Manage Books use case.....	21
Table 3.6 Use case narrative for Add Books use case .....	22
Table 3.7 Use case narrative for View Transactions use case.....	23
Table 3.8 Use case narrative for Pay Credit use case.....	24
Table 3.9 Use case narrative for Generate Report use case.....	25
Table 3.10 Use case narrative for Manage Users use case .....	27
Table 3.11 Access control for modules.....	29
Table 4.1 Implementation Environment .....	33
Table 5.1 Test cases for the Purchase Module.....	45
Table 5.2 Test cases for the Sales Module.....	46
Table 5.3 Test cases for the Stock Module.....	48
Table 5.4 Test cases for the Finance Module.....	49
Table 5.5 Test cases for the Web Module.....	49
Table 5.6 Test cases for the Report Module.....	50
Table 5.7 Test cases for the Admin Module.....	51
Table A.1 Recommended Hardware Requirements.....	57
Table A.2 Recommended Software Requirements.....	57
Table B.1 Use case narrative for View Purchase .....	60
Table B.2 Use case narrative for View Sales.....	62
Table B.3 Use case narrative for Manage Dealers.....	64
Table B.4 Use case narrative for Receive debit.....	65
Table B.5 Use case narrative for Place order .....	65
Table B.6 Use case narrative for Generate Sales report.....	66
Table B.7 Use case narrative for Generate Stock System.....	66
Table B.8 Use case narrative for Backup Data.....	67

Table B.9 Use case narrative for edit Profile.....	67
Table B.10 Use case narrative for Admin Supplies.....	68
Table E.1 Test results for the Purchase Module.....	86
Table E.2 Test results for the Sales Module.....	87
Table E.3 Test results for the Stock Module.....	88
Table E.4 Test results for the Report Module.....	89
Table E.5 Test results for the Administration Module.....	90
Table E.6 Sample questionnaire.....	91
Table E.6 User Evaluation Feedback Results.....	93

# List of Acronyms

CI - CodeIgniter  
CRUD - Create, Read, Update and Delete  
CSS - Cascading Style Sheet  
DBMS - Data Base Management System  
HTML - Hyper Text Markup Language  
JAD - Joint Application Development  
MVC - Model-View-Controller  
OOD - Object Oriented Designing  
OOP - Object Oriented Programming  
PHP - Php-Hypertext Preprocessor  
PDF - Portable Document Format  
WBBM –Web Based Book Management  
WB – Wisdom Bookshop  
SKU – Stock keep Unit  
RAD - Rapid Application Development  
RAM - Random Access Memory  
RUP - Rational Unified Process  
SDLC - System Development Life Cycle  
SQL - Structured Query Language  
UML - Unified Modeling Language  
URL - Uniform Resource Locator  
WWW – World Wide Web

# Chapter 1 | Introduction

## 1.1 Introduction

The future of any business totally depends on how efficiently they can make the business critical decisions, in which a well automated information system plays a vital role. Application of information technology is essential in the development of such a system as well. Because of that, most of the businesses use automated information systems back-up by various IT tools in order to process information accurately and efficiently as needed in their business decision making. This chapter provides a clear picture of the client, motivations, objectives and scope of the project.

## 1.2 The Client

WISDOM BOOKSHOP was established in 1994 at Wattala with the objective of serving the educated industry which only handle retail customer's books sales at the begin for limited customers who came to bookshop. Specializing as a for all your educational needs of both international and local curriculums. We continue to grow the changing needs of our industry but the business was able to expand their service to other related areas within a year after the founding.

“World Class book shop for Schools and universities and to provide an exclusive shopping experience to our customers” which are the vision is to serve over 300 loyal customers and consists of well experienced and energetic professionals. They are dedicated to customer satisfaction through best products, online shopping experience and other delivery. The service provide by the business can be categorized as Local and International text books, other books, stationary and other educational products. Bookshop Service provide to their customers in all levels like as Primary, Secondary, Local, London (O/L, A/L) and other International Exams.

WB has acquired a good reputation for providing a quality in whatever it ventures into and perfect services to the customers - friendly environment, and also shown a high progress and expects to expand their services further. Presently plays a vital role with International and Local Examination text books in Bookshop Industry.

## 1.3 Motivation

The existing system transactions are not performing properly and overcome some of the issues in entire system modules such as Purchase, Sales, Stock Management and Report generations are inaccurate, time wasting, inefficient & unreliability of the semi-automated system due to increase of new customers and sales through existing website of wisdom.

The above mentioned issues in existing system, so need to eliminate therefore Bookstore Management for Wisdom taken decision to make fully computerized the system then it will be help to provide better customer solution and lead the business to a success in a competitive business market. These problems occur during last years because of the large number of data volume processed due to increase of new customers, books and branches.

Considering this problem, proposed a web based solution to bookstore management system for Wisdom to reach the finest level to fulfill the quality & standard of requirement to the customer's due to unavailability of new books in the store operations. This solution is built in a way to cover the overall administration process of store within all modules.

Finally, there is a special interest and willingness to make use of the web based technologies and development of web applications. Through this, an opportunity can be gained to practice the knowledge of the theories and techniques studied under BIT program me as well.

## 1.4 Objectives

The overall objective of the project is to enhance and improve the effectiveness and efficiency of the service provided by the business by introducing a full web based system which can be divided further as follows,

- To improve the efficiency of daily business process such as purchasing, sales and stock controlling by eliminating the considerable delays and bottlenecks.
- To enhance the accuracy of the business process such as Purchase, Sales, Stock Management via automated WBBM.

- To eliminate the delay and inaccuracy in providing details needed to the external parties such as customers and suppliers, through which customer-business-supplier relationship can be strengthened.
- To enhance the business decision making process by providing the necessary reports timely when needed.
- To preserve the historical business data related to purchasing and sales.
- To decrease the work load and responsibilities by decentralizing the work load and responsibilities among the staff members.
- To eliminate unnecessary paper working & workload in the daily business activities.
- To enhance the security and recoverability of the business-critical data by introducing critical security features such as access control mechanism, authentication and accountability.

## 1.5 Scope

Scope of the project describes the area covered by the project that was determined by balancing the time and the budget with the objectives of the process. The scope of the project can be summarized as follows,

- **Stock Management** – includes maintaining stock levels, re-order status, notifying books which have low level, managing book categories and searching facility.
- **Purchase Management** – includes managing requisitions, making purchase orders, processing purchases, handling purchase returns and managing agents.
- **Sales Management** – includes managing retail sales and whole sales, handling sales returns and managing dealers.
- **Financial Management** - to manage the credits and debits related to the agents and dealers respectively, with the payment history of the purchases and wholesales.
- **Web Module** - represents the frontend website of the WBBM tasks handled by this are placing orders online, viewing orders history, replying to inquiries and so on.
- **Reports Module** – includes generating sales, purchasing and stock reports and generating business process related documents such as customer invoice, purchase orders and so on.

- **Administration Module** – includes managing system users, controlling access levels, manage system log and generating data backup.

## 1.6 Structure of the Dissertation

The dissertation consists of six main chapters. A number of appendices have been attached at the end in order to provide some additional information as well. A brief description of the contents of the main chapters has been given below.

The *Analysis phase* explains about the detail overview of the analysis, requirements gathering techniques, study of the existing system and functional, non-functional requirements and literature review are discussed under this chapter.

The *Design phase* includes the Introduction to design, overview of the design methodologies, alternative solutions, complete detailed design of the proposed system, tools and techniques and Database Designs are discussed under this chapter. This chapter also includes detailed use-case diagrams, class diagrams, sequence diagram, activity diagram and deploy diagrams. The chapter gives the features of the Graphical User Interfaces (GUI), which help the users to interact with the System using the screen shots.

The *Implementation phase* describes the development procedures of the system. Further implementation environment, software tools and technologies used in the implementation, application structure and code segments of the system.

This *Testing and evaluation phase* defines how the system was tested using various sample data methods, test plans and the outcomes of the assessment. This includes testing procedures and test cases.

The *Conclusion* is the final chapter of the dissertation, which includes the critical evaluation of the project, review the entire development process carried out and gives details about the recommended future improvements and lessons learnt throughout the project have been discussed under this chapter.



# Chapter 2 | Analysis

## 2.1 Introduction

Analysis is one of the crucial phases in the software development process in which the problem domain is studied in detail along with pros and cons of the existing system and the requirements of the proposed system are clearly identified.

This chapter describes the fact finding techniques, overview of the existing system, requirements of the proposed system, functional and non-functional requirements and literature review in detail. The main processes of this phase include domain understanding, current system process is going on, draw backs of the present system, requirements collection of the proposed system and similar system reviews. [1]

## 2.2 Fact Finding Techniques

**Interviewing** is the most popular technique for gathering requirement that enables the interviewer to clarify the facts further as needed, but time consuming and costly. **Observation** is very reliable and commonly used to confirm the requirements gathered but can be made some embarrassment for the staff. **Examining the existing documentations** enables analyst to discover some background information which have not been revealed yet. **Questionnaires** are used often especially in the situation where the number of users are very high and probably spread over the wide area. [25]

It is more practical to use a combination of multiple techniques rather than one particular technique for requirement gathering process. Interviewing, on site observation and examining the existing documents were used to discover requirements for the WBBM.

Several interviews were conduct with the managing director, manager, suppliers, customers and the internal staff with combinations of close and open-ended questions. Facts were primarily gathered by interviewing the manager who is the primary user of the proposed system.

The daily business processes such as sales, purchase are observed in couple of sessions. The captured facts were represented as scenarios to give a clear picture on basic requirements and interaction sessions of the existing system documents were

examined in order to extract the business details needed to store in the system and the format of the documents. The issues in the documentations were also noticed.

The staff members in different management levels were selected to discover requirements related to the Management and Administrative procedures. It can be a beneficial to us, when we intend to acquire more knowledge and understanding about the domain.

## 2.3 The Existing System

The existing system of the WBBM is a poor semi-automated computer system. The existing system was divided into subsystems as follows in order to study the system more efficiently.

- Purchase Management
- Sales Management
- Stock Management (Inventory)
- HR

### 2.3.1 Purchase Management

This fully depends on book agents supplied from outside, the main purchase of the stores is the book in which the particular book is always purchased by the one particular agent. The agent can be either the publisher or the local distributor. The book brought by the suppliers weekly or monthly is checked by manager before purchase, including the supplier details

### 2.3.2 Sales Management

The store sales can be divided into two categories, Wholesales & Retail sales. In Retail Sales, the produced book is sold directly to the customers in small quantities. The Sales Management System handles all the details related to the whole sales and sales returns from both types of sales.

At present, the manual system updates the stock according to sales, and returns

### 2.3.3 Stock Management

The stock is managed by the manager. The current stock is updated when whole sales or dispatches for damage sales take place. Further, the stock is updated when there are sales returns. The stock from the warehouse is sent to sales section in FIFO (First-In, First- Out) method. All information is recordings and updates are done manually.

### 2.3.4 HR Management

All the current staff members are handled in the Staff worksheet by the manager. Details of the past staff members are removed from the worksheet periodically as well. Each designation has a number of annual leave, no-pay rate and over-time rate. Leave taken by the staff members and their over-time details are recorded in the Staff Leave & OT book by the manager. The bank deposit list is also prepared in order to transfer the salaries to the bank accounts of each employee. EPF & ETF are also transferred.

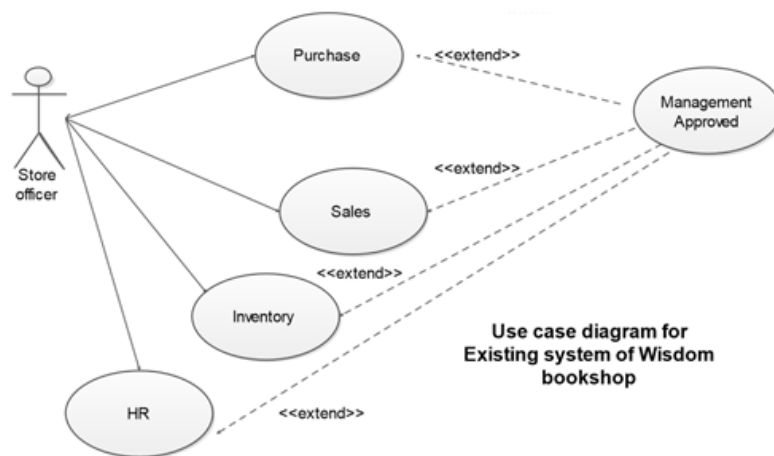


Figure 2.1 Existing Bookshop System for wisdom

Additionally, Wisdom has their own website in which the customers are able to search the books and place the orders. The orders are entered in to the internal system as usual retail order when process. The above Figure 2.1 shows the existing system of WBBM.

## 2.4 Drawbacks of the Existing System

The following major drawbacks have been identified in the existing manual system.

- The historical data of the business such as the detail of the past sales and purchases, are not preserved in the system. Removing historical data produces some inconsistency to other existing data as well.
- Details of all the dealers and the agents are not kept in the system which reduces the benefit of the data in business analyzing processes. Only the summary of the daily transactions is stored in the system as well.
- Most of the data is entered into the system twice, firstly recorded in the books. This consumes time, produces unnecessary workload and causes the unavailability of the updated data during the business hours.
- There is no clearly defined way in preparing purchasing orders, which occurs bottlenecks in purchasing process.
- Sensitive business critical data is exposed for anyone and there is no backup routine procedure which increases the vulnerability of the security of the data.
- Critical calculations are done in manually
- Hand written contract documents used which are inconsistent in format.
- Complex monitoring of business progress

## 2.5 Functional Requirements

“In Software Engineering, a functional requirement defines a function of a software system or its component. Functional requirements describe what should be done by the system. A function is described as a set of inputs, the behavior, and outputs.” [2]

When it comes to functional requirements, it states how the system reacts to a particular input and what kind of an output or process is carried out through the system.

Following are a list of functional requirements that are identified WBBM during the analysis stage:

## Purchase Management

- The system should generate unique identification numbers automatically in order to eliminate the duplication in the entities such as purchase orders and purchases.
- The system should be able to manage purchase requisitions.
- The system should compile purchase orders on the demand of the staff member.
- The system should be able to generate purchasing related documentations such as purchase orders and GRN.
- The manager should be able to manage the details of the agents.
- The system should be able to process purchase returns with generating related documents.

## Sales Management

- The manager should be able to manage the dealer's information.
- The operator should be able to search books with related details such as price availability and so on when needed.
- The system should be able to process direct sales and generate invoices on the demand of the staff.
- The user should be able to maintain information related to both, wholesale and retail sales.
- The system should process sales return and manage the relevant details as well.

## Stock Management

- The System should allow the manager to manage the stock category list
- The system should manage the books with their related details.
- The system should update the stock, when sales, dispatch or waste disposal has been carried out.
- This also should notify the low stock levels.

## Financial Management

- System should be able to manage the credits and debits of agents and dealers.
- System should manage the payment history of the purchases and wholesales.
- System should be able to process the payments and cheque returns.

## Reports Module

- The system should allow the managers to generate and view the timely reports related to Sales, Purchase and Stock.
- The system should allow the users to print the Reports and documents.

## System Administration Module

- The administrator should be able to deactivate or re-activate the users if needed.
- The administrator should be able to reset the passwords of the users if needed.
- The system should generate data backup on the demand of the administrator.

## 2.6 Non-Functional Requirements

“In Systems Engineering and Requirements Engineering, a non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors.” [3]

Success of this system is not only dependent on the functional requirements; it is also based on the non-functional requirements. It acts as a main role for the success of the system.

Non-functional requirements describe the characteristics or constraints the system should have, which affect the whole system rather than a specific part. The non-functional requirements of the WBBM have been described below.

- The system should be user-friendly. System should have a consistency in both its functionality and appearance. The staffs of the WB are very familiar with windows environment. Therefore, the system should be close to windows GUI environment as possible. The system should also be easy-to-learning.
- The system should be secure. The system will handle sensitive business critical data. Hence enough security should be imposed. User authentication and access controls should be provided in the system.
- The system should be reliable. The system should provide data validation procedures in order to ensure the accuracy of the data. System should also be available at any given time and should work with minimum failure rate.
- The system should be portable. The system will be accessed by the customers and suppliers using their own systems which can have different platforms. Therefore, WBBM should run perfectly on any popular platforms.
- Efficiency of the System, including response time and processing time should be less, when customer order a book online or searching a book since most of the users expect this feature from a System.

## 2.7 Literature Review

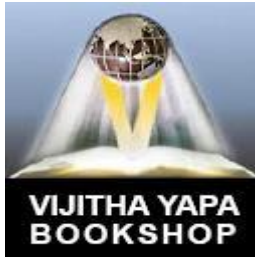
Studying the existing similar systems regarding their purpose, features, advantages and disadvantages would be helpful as well. Following systems and applications were studied to get some ideas regarding the development of WBBM.

Wisdom system was the primary classification to understand the background which we have to work and to understand the concepts and procedures of a bookstore system.



Once an invoice is synced from SalesBinder, if you delete that invoice, it will also be removed. This works for invoices, however deleting synced customers in SalesBinder does not delete them. It

doesn't support deleting customers from integration. Lastly, resetting an invoice in SalesBinder will delete all synced payments towards that invoice as well. Create all the attributes you need and use them on as many different inventory items as you like. Setup your attributes as plain text, unique text, or even a pre-populated select menu where you can define what options are available. [4]



Vijitha Yapa Group establishes a chain of bookshops, and there is lot of branches. Vijitha Yapa Bookshop is the largest bookstore chain in Sri Lanka and offers more than 250,000 titles altogether. The Vijitha Yapa group holds the exclusive rights to sell books in the transit lounge of Colombo's International Book fair also. Vijitha Yapa Subscriptions is the largest subscription agent in Sri Lanka. [5]



The use of technology and software tools to meet specific customer requirements, Sarasa Soft Solutions has carved a niche all of its own, providing enterprise software solutions to large and small scale retail organizations. In your office or warehouse, we have the most modern technologies to maximize efficiency and precision in whatever area of business you are engaged. I refer Point of Sales (POS) Systems, Sales, Inventory Management Systems, and Cash Register. IT offers complete customization to identifying customer requirements to designing specific software to cater to their every need and the support of a personalized service that is second to none. With the use of state-of-the-art technology and software tools to meet specific customer requirements, Sarasa Soft Solutions has carved a niche all of its own, providing enterprise software solutions to large and small scale retail organizations across the country. [6]



The Core activities are retail technologies, Inventory control, Sales control, Payroll, Attendance, and Accounting. It uses standalone machines and POS terminals for the retail business. A cash register uses an electronic device for calculating and recording sales transactions as well as stock control, cash drawer and print a receipt for the customers. [7]





When you enable the QBO integration, it remembers that point in time and only syncs newly created invoices after that

moment to ensure you don't end up duplicating previous invoices in QBO. Inventory data does not sync your inventory list into QBO. This is simply because QBO doesn't support all the inventory functionality does and it would unnecessarily duplicate functionality. Instead, QBO will send over invoice line-items as a generic "Inventory Item" with all the relevant details) in the description field. [8]

The following table 2.1 summarizes the facilities provided by the above systems

Modules	SalesBinder	Vijitha	Sarasa Soft	Retail IT	QBO
<b>Administration</b>					
<b>Stock</b>	√		√	√	
<b>Sales</b>	√	√	√	√	
<b>Purchase</b>	√	√	√	√	
<b>Financial</b>					√
<b>Report</b>					√
<b>Home - profile</b>		√	√		

Table 2.1 Literature Review

## 2.8 Development Methodology

Software development methodology is a pre-defined approach for the development of software that consists of phases, best practices, deliverables and tools in order to produce quality software within the time and financial constraints.

Waterfall model describes the entire process as a set of sequential phases which is very simple and well-documented model, but not suitable if the requirements are not clear.

Evolutionary prototyping is a combination of iterative development and the prototyping which is perfectly suitable for the situations where the requirements are not clear. RAD is an iterative development methodology with the intention of providing the solution as soon as possible which consists of a set of technologies and tools such as prototyping and JAD.

RUP is an iterative and use case-driven software development methodology introduced by IBM which consists of principles, best practice as and tools.” [21]

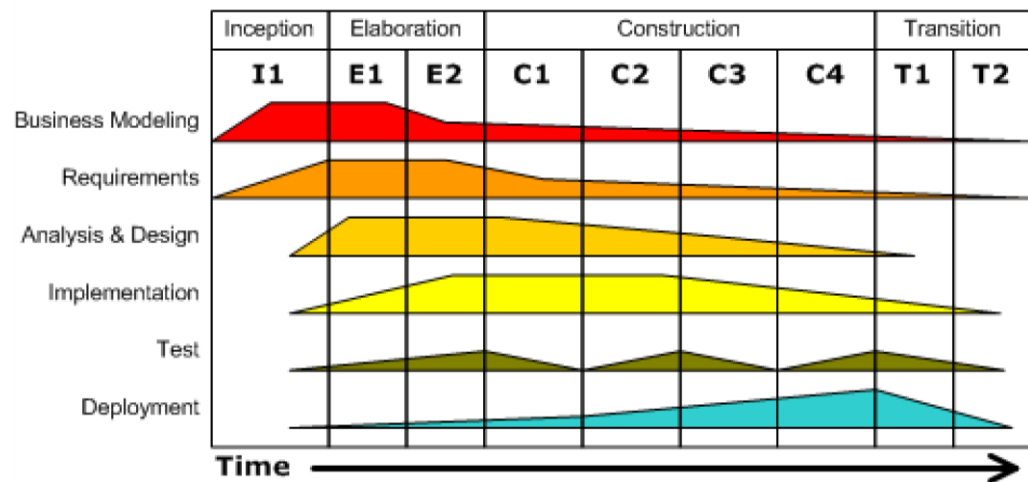


Figure 2.2 Two Dimensional view of RUP. [9]

As shown in figure 2.2, it describes two perspectives called static view and dynamic view. Static view describes a set of work flows while dynamic view describes four phases each of which has own objectives and milestones.

Inception phase focuses on the scope, schedule and budget, elaboration phase focuses on the analysis, construction phase focuses the implementation and transition phase focuses on the delivering the system to the users. [23]

RUP was selected as the methodology of the WBBM because the features of it are well suited with the development nature of the WBBM. Some of those features are,

- Use-case driven approach with the integration of UML tools.
- Risk analysis of early stages reducing the critical issues can be occurred at the middle of the development.
- Promoting iterative development, prototyping and always focused on the working software rather than the documentations.
- Requirements can be changed during the process; RUP can manage this changing requirements.
- RUP uses UML (Unified Modeling Language) models to address the static view as well as the dynamic view of the system.
- Throughout the development it ensures the software quality as well as the standard.
- Switching into previous lifecycle stages within the development process can be achieved.

# Chapter 3 | Design

## 3.1 Introduction

Design phase of the system development process makes the most suitable solution for the underlined problem which is clearly identified by the analysis phase. At the end of the design phase the complete blueprint of the proposed system is resulted.

This chapter describes the common software development methodologies with selected methodology for the proposed system, alternative solutions with the solution selected, design techniques and the complete design of the WBBM.

## 3.2 Development Strategy

Development strategy is the way to follow in fulfilling the identified requirements. There are several strategies taken in to the account regarding the WBBM.

Purchasing a set of commercial software can be reduced the development cost & time and provided some extra quality, but may have to be paid for some unnecessary features, may not fit perfectly for all the requirement and lack of configurable. Standalone systems provide advantages such as easy to implement and maintain, reduced incompatibilities and security vulnerabilities, but unable to serve to the remote users.

Development of web based system from scratch was selected for WBBM as the development strategy.

Developing the software from scratch results a software which is perfectly matched with the identified requirements. And also a highly configurable and upgradeable product is produced with minimum incompatibility issues. Web based system enables to serve remote users which are essential in order to fulfill the requirements such as allowing the customers to place orders, searching books and tracking down the order progress.

## 3.3 Reasons to Choose the Web Based System

- The client particularly requested for a Web Based System.
- Easy to monitor the stock processes and generate informative reports from any location.

- Maintenance is easy, because the database is centralized and everything is synchronized.
- Easy Deployment.
- Can be implemented on Client-Server Architecture and use a web-browser as the client interface
- The user is simply redirected to appropriate pages.
- The adaptation of MVC makes the future enhancements and maintenances straight forward.
- A web based system to automate day to day activities had been requested by the client.
- The cost of hardware as well as software can be reduced by the implementation of a web based solution.
- Web systems are platform independent and any different platform can be used.
- Access to the system can be obtained from anywhere without reaching to the office premises for that particular activity.
- When comparing with stand-alone systems, maintaining a web system is simple.

### 3.4 Design Techniques

Design techniques are the well-defined approaches for the purpose of system design and modeling. Each of them has own techniques, features, advantages and disadvantages. Therefore, it should be looked close before selecting a particular approach.

Structured Design Technique (SDT) is a data-flow oriented approach in which the system is described as a set of functions. Various diagrams are used in SDT such as Context diagrams to show the highest level view of the system, Structure chart to show the top down decomposition of the system and DFD to depict the flow of the data through the system.

Object Oriented Design (OOD) is the most popular system design approach in the present in which the system is described as a collection of objects. OOD uses the concept of object orientation such as classes & objects, inheritance, encapsulation and

generalization for the design. It is well integrated with UML which is a powerful modeling language provides a rich collection of diagrams to model the system form various perspectives. Using OOD also enables to use OOP directly in the implementation. [10]

OOD with UML were selected for the WBBM as the design technique because of its features and advantages described above. Use case diagrams, Class diagrams, Activity diagrams and Sequence diagrams were used to model the functionalities of the system, static structure of the system objects, flow of the activities in a particular process and interactions of the objects in carrying out a particular process respectively.

### 3.5 Design of the WBBM

WBBM is divided in to seven modules as follows in order to be efficient the design process and further implementation.

- Purchase Module
- Sales Module
- Stock Module
- Financial Module
- Reports Module
- Web Module
- Administration Module

The top-level use case diagram of each module is represented in the following sub sections. Some use case narratives and additional diagrams are also provided when needed.

Please refer the appendix B – design documentation for other important use case narratives and the additional diagrams.

#### 3.5.1 Purchase Module

This module only can operate manager and computer operator. This section involves, requisitions purchase orders, purchase, purchase return orders, purchase returns. Requisitions purchase orders are handling by manager as completed or cancel

likewise. Manager can make a simple note for a particular rare book requirement ask by a customer or dealer. The managers examine the pending requisitions before a new purchase order, and then check their availability and if it's OK, they can add it as a new book and then order. We cannot use the requisitions directly for the purchase order process, because the details provided by the customer may vary and the availability or the agent is unknown, hence should be verified before adding them as a new book.

The following figure 3.1 use case diagram shows the main activities in Purchase module.

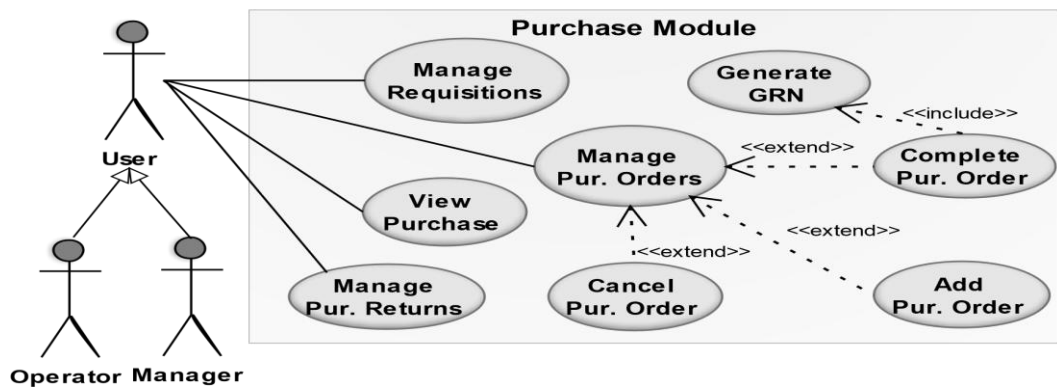


Figure 3.1 Use case diagram for the Purchase Module

Use case Name	Manage Purchase Orders
Actors	Operator, Manager
Description	Actor manages the purchase orders.
Pre-conditions	Actor must log in to the system.
Main Flow	
<ol style="list-style-type: none"> <li>1. Actor navigates to the 'Purchase' section.</li> <li>2. System displays the section.</li> <li>3. System provides options to add, complete and cancel purchase orders.</li> <li>4. System updates and confirms.</li> </ol>	
Alternate Flows	
<p><b>3.1. Actor adds a purchase order.</b>                      3.1.1. "Add purchase order" use case is executed.</p> <p><b>3.2. Actor completes a purchase order.</b>                      3.2.1. "Complete purchase order" use case is executed.</p> <p><b>3.3. Actor cancels a purchase order.</b>                      3.3.1. "Cancel purchase order" use case is executed.</p>	
Post-conditions	none

Table 3.1 Use case narrative for the Manage Purchase Orders use case

Use case Name	Add Purchase Order
Actors	none
Description	Actor places a purchase order.
Pre-conditions	The use case “Manage Purchase Order” must have been executed.
Main Flow	
<ol style="list-style-type: none"> <li>1. Actor navigates “New Purchase Order” sub suction.</li> <li>2. System displays the form to fill.</li> <li>3. Actor selects the agent.</li> <li>4. System updates the screen with low stock books.</li> <li>5. Actor selects the books to order, fills the other relevant data.</li> <li>6. System validates the inputs</li> <li>7. System adds the purchase order and confirms.</li> </ol>	
Alternate Flows	
<b>4.1 One or more field is invalid.</b> <ol style="list-style-type: none"> <li>4.1.1 System informs the error(s) and prompts to correct.</li> </ol>	
Post-conditions	Purchase order has been added successfully.

Table 3.2 Use case narrative for the Add Purchase Orders use case

### 3.5.2 Sales Module

Sales module involves Add sales – retail, whole sales, View sales –detail view, sales returns. In Add Sales operate new retail sales, new wholesale, whole sales made on credit, if the dealer pay at the moment should add as “receive debit”. The cash amount paid should be added, because to print the balance in the invoice as usual, although no stored in the system. It will be warned when calculating. The discontinued books are also allowed to be sold. Because the status of the book affected for purchasing. Invoice printed in both cases.

View Sale(s) which has credits and not settled after a month shown in red highlighted. In the sales details of a particular sale. Allows searching book, very similar to “Books” in stock section. But eliminate unnecessary ones, such as min-stock, max-stock etc. because here the purpose is to check book on the demand of the customers. The sales returns involve only the quantity, not the financial side, because the price difference is very rare and ignored if any.

The following figure 3.2 use case diagram shows the main activities in Sales module.

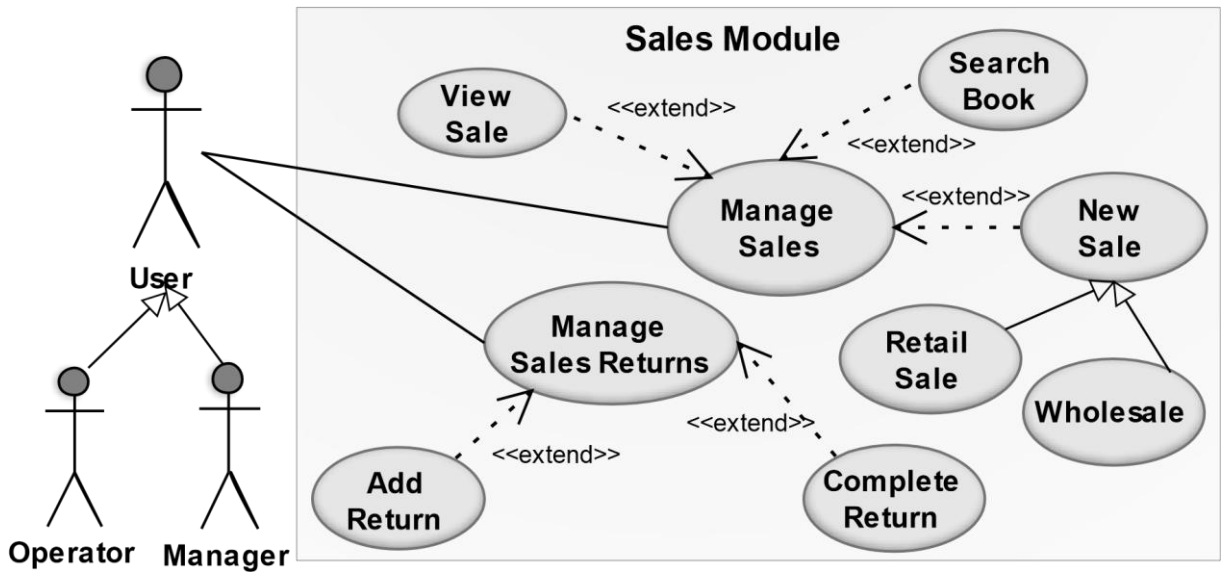


Figure 3.2 Use case diagram for the Sales Module

Use case Name	Manage Sales
Actors	Operator, Manager
Description	Actor manages the sales.
Pre-conditions	Actor must logs in to the system
Main Flow	
<ol style="list-style-type: none"> <li>1. Actor navigates to ‘Sales’ section.</li> <li>2. System provides options to add, view sales and search a book.</li> </ol>	
Alternate Flows	
<p><b>2.1 Actor places a new sale.</b>                  2.1.1 ‘New Sale’ use case is executed.</p> <p><b>2.2 Actor Views a sale.</b>                  2.1.2 ‘View Sale’ use case is executed.</p> <p><b>2.3 Actor Searches a book.</b>                  2.3.1 ‘Search Book’ use case is executed.</p>	
Post-conditions	None

Table 3.3 Use case narrative for Manage Sales use case

Use case Name	Search Book
Actors	none
Description	Actor Searches a book.
Pre-conditions	The use case “Manage Sales” must have been executed.
Main Flow	
<ol style="list-style-type: none"> <li>1. Actor navigates to ‘Search Book’ sub section.</li> <li>2. Actor specifies the searching Criteria.</li> </ol>	



3. System searches the book(s) according to the criteria specified.	
4. System displays the result.	
Alternate Flows	
<b>4. There is no searching result.</b>	
4.1 System informs about the empty result.	
Post-conditions	none

Table 3.4 Use case narrative for Search Book use case

### 3.5.3 Stock Module

This module involves, Categories, Agents - person or organization who supply the books, Dealers – person / organization to whom the books sold, Books - book details and stock details. Each book has one category. This cannot be edited later. Each book is identified by Stock keep unit (SKU) that is standard term for product ID, unique code used to identify products in inventory control, also used in wisdom’s web site. Add new book ISBN and agent combination is unique. Means you can assign similar ISBN for more than one agent, because the system uses SKU.

Cancelled categories and past agents are shown in red-text. The books in which the current stock is less than or equal to its min-stock are shown in red-highlighted. The availability of the functionalities, such as edit / discontinued are based on the current status. This is common for other sections. E.g. you cannot edit discontinued book, or the contact details of a past agent.

The following figure 3.3 use case diagram shows the main activities in Stock module.

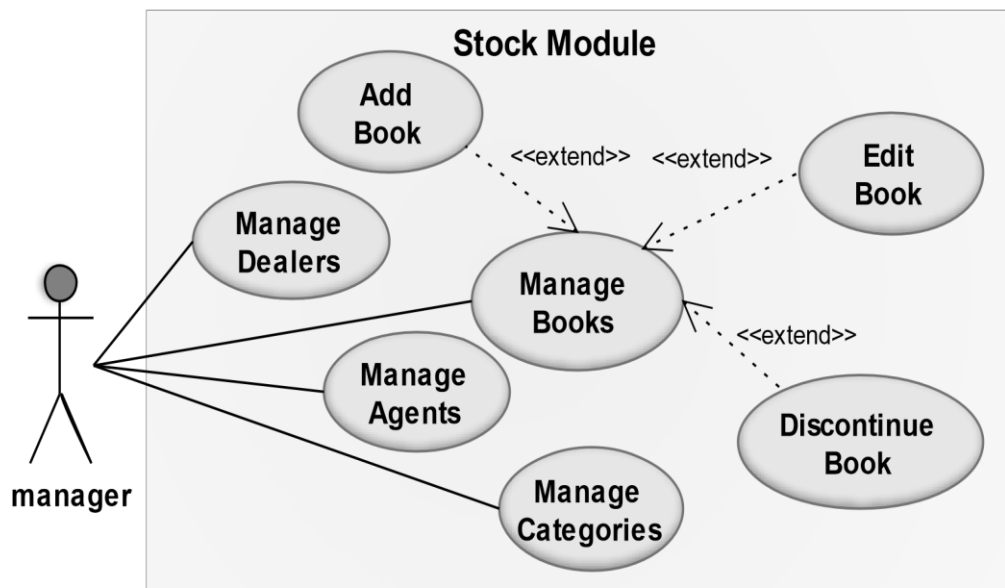


Figure 3.3 Use case diagram for the Stock Module

Use case Name	Manage Books
Actors	Manager
Description	Manager manages the books' details.
Pre-conditions	Manager must log in to the system.
Main Flow	
<ol style="list-style-type: none"> <li>1. Manager navigates to the 'Stock' section.</li> <li>2. System displays the section with the books details.</li> <li>3. System provides options to the add, edit discontinue a book.</li> </ol>	
Alternate Flows	
<p><b>3.1 Manager adds a new book.</b>  3.1.1 "Add Book" use case is executed.</p> <p><b>3.2 Manager edits an existing book.</b>  3.2.1 "Edit Book" use case is executed.</p> <p><b>3.3 Manager discontinues an existing book.</b>  3.3.1 "Discontinue Book" use case is executed.</p>	
Post-conditions	none

Table 3.5 Use case narrative for Manage Books use case

Use case Name	Add Book
Actors	none
Description	Manager adds a new book.
Pre-conditions	The use case "Manage Books" must have been executed.
Main Flow	
<ol style="list-style-type: none"> <li>1. Manager navigates to 'New Book' sub section.</li> <li>2. System displays a form to fill.</li> <li>3. Staff fills the form and sends.</li> <li>4. System validates the inputs.</li> <li>5. System verifies that the book is not already existed.</li> <li>6. System adds the book and conforms.</li> </ol>	
Alternate Flows	
<p><b>2.1 Manager needs to add a new Category.</b>  2.1.1. "Manage Categories" use case is executed.</p> <p><b>2.2. Manager needs to add a new Agent.</b>  2.2.1. "Manage Agents" use case is executed.</p> <p><b>3.1. One or more field is invalid.</b>  3.1.1. System shows error(s) and prompts to correct.</p> <p><b>5.1 The book is already existed.</b>  5.1.1 System informs to correct and continue.</p>	
Post-conditions	The new book is added successfully.

Table 3.6 Use case narrative for Add Books use case

### 3.5.4 Financial Module

The purpose of this module is to just managing the credits and debits of the Dealers and Agents respectively related to the wholesales and purchases not to monitor the monetary status of the business. So it not follows any standard book keeping rules. Retail Sales made in cash on the time. Wholesales made on debits. Purchase made on credits. Paying credits to the Agent or receiving debits form the Dealer usually happens as bulk not one by one purchase / sales. It's in cash or by bank transferring or cheque.

The following figure 3.4 use case diagram shows the activities in Finance module.

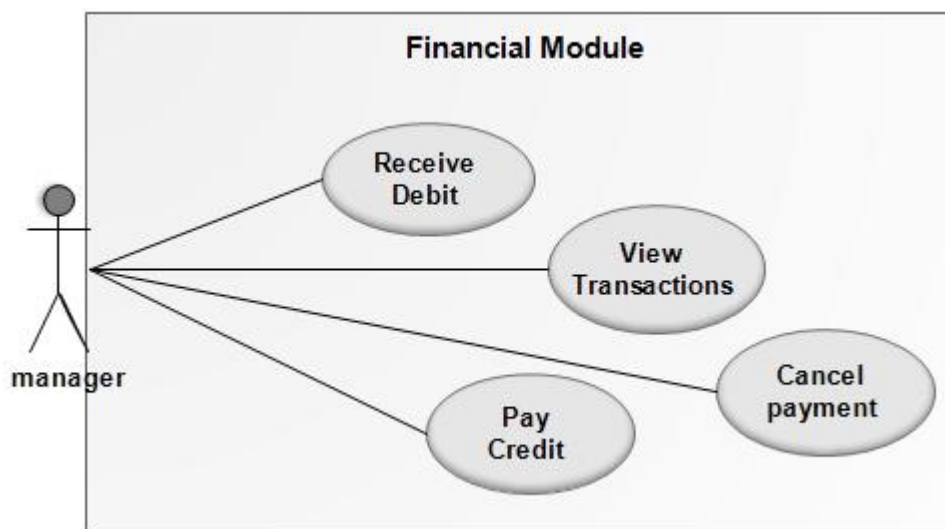


Figure 3.4 Use case diagram for the Financial Module

Use case Name	View Transactions
Actors	Manager
Description	Manager views the transactions.
Pre-conditions	Manager must log in to the system.
Main Flow	
1. Manager navigates to the 'Financial' section. 2. System displays the section with the recent transaction details.	
Alternate Flows	
None	
Post-conditions	none

Table 3.7 Use case narrative for View Transactions use case

Use case Name	Pay Credit
Actors	Manager
Description	Manager adds a credit paid to an agent(creditor).
Pre-conditions	Manager must log in to the system.
Main Flow	
<ol style="list-style-type: none"> <li>1. Manager navigates to ‘Creditors &amp; debtors’ sub section.</li> <li>2. System displays list of creditors and debtors with amount.</li> <li>3. Manager clicks pay credit button in front of particular creditor.</li> <li>4. System displays a form to fill.</li> <li>5. Manager fills and submit.</li> <li>6. System validates, updates the balance and confirms.</li> </ol>	
Alternate Flows	
<p><b>6.1 One or more field is invalid.</b></p> <p>6.1.1 System shows error(s) and prompts to correct.</p>	
Post-conditions	The credit is added successfully.

Table 3.8 Use case narrative for Pay Credit use case

### 3.5.5 Web Module

The Web module represents the frontend of the WBBM which is responsible for facilitates the customers and the suppliers to interact with the business. Some of the tasks handled by this are placing orders online, viewing orders history and so on.

The following figure 3.5 use case diagram shows the main activities in Web module.

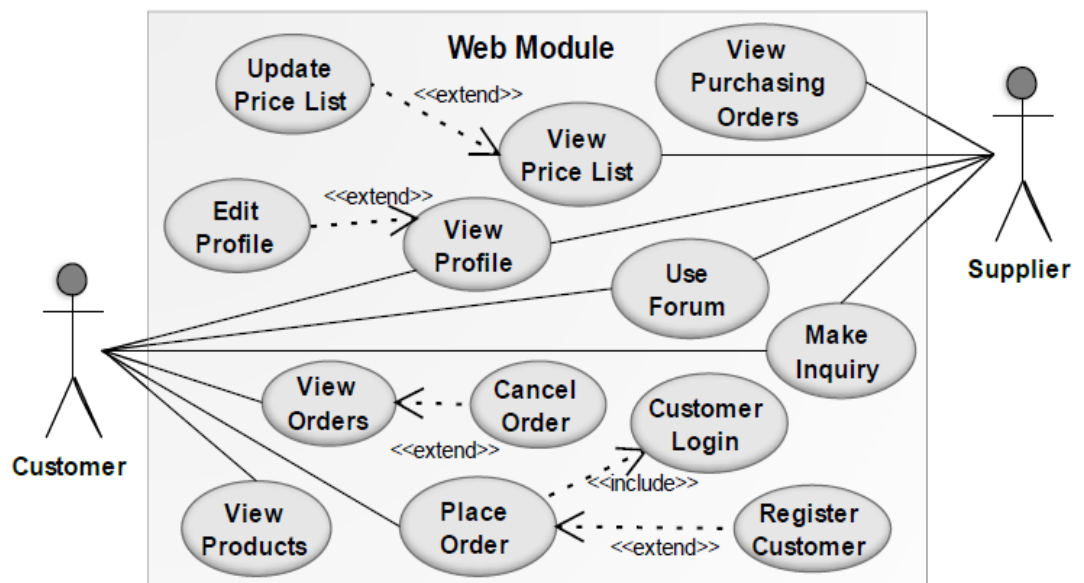


Figure 3.5 Use case diagram for the Web Module

### 3.5.6 Reports Module

This includes generating sales, purchasing and stock reports generating business process related documents such as sales invoice, purchase orders and so on. Reports can be either hard copy or on-screen tables. Report generating Module is only for authorized users according to their needs. They have a lot of variations according to the criteria, month and year as shown in *reports* screen. The inactive options are shown in red texts in the list in general.

The following figure 3.6 use case diagram shows the main activities in Report module.

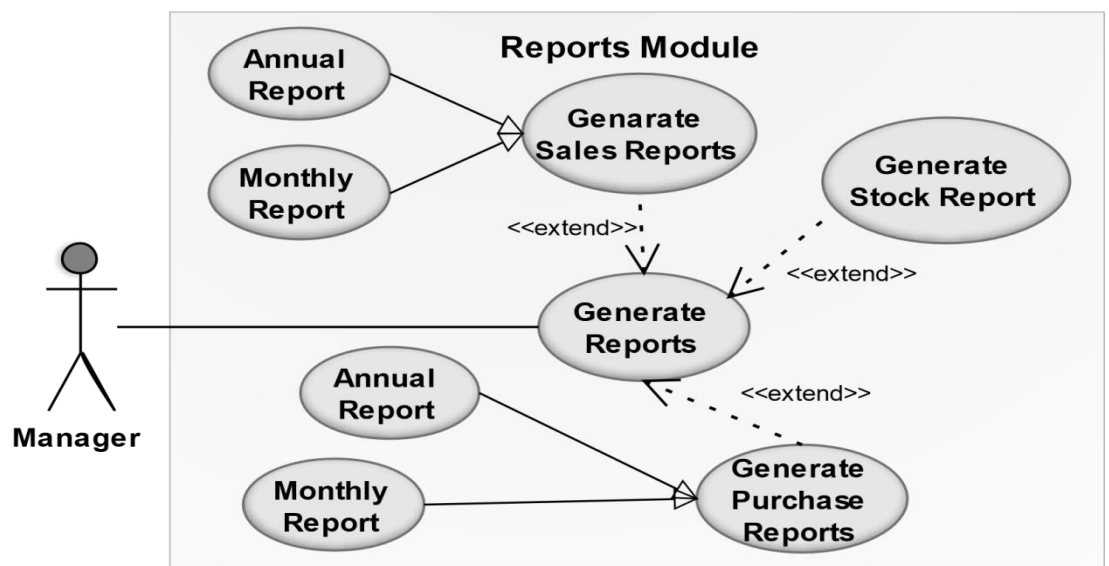


Figure 3.6 Use case diagram for the Report Module

Use case Name	Generate Reports
Actors	Manager
Description	Manager generates the relevant reports.
Pre-conditions	Manager must log in to the system.
Main Flow	
<ol style="list-style-type: none"> <li>1. Manager navigates to the 'Reports' section.</li> <li>2. System displays the section.</li> <li>3. System provides options to generate various reports.</li> <li>4. System generates the particular report.</li> </ol>	
Alternate Flows	
<p><b>3.4. Manager generates sales report.</b>          3.4.1. "Generate Sales Reports" use case is executed.</p> <p><b>3.5. Manager generates purchase report.</b></p>	

3.5.1. “Generate Purchase Reports” use case is executed.	
<b>3.6. Manager generates stock report.</b>	
3.6.1. “Generate Stock Report” use case is executed.	
Post-conditions	none

Table 3.9 Use case narrative for Generate Report use case

### 3.5.7 Administration Module

This module intends for administering the WBBM itself. This includes managing system users, controlling access levels, manage system log and generating data backup. The administrator is able to deactivate or re-activate the users if needed. The administrator is also able to reset the passwords of the users if needed. The data backup is generated on the demand of the administrator as well.

Super admin is a virtual admin with full access for all modules. However only recommend to create new admin. Admin creates operators and managers. Super admin account cannot be edited, reactivated, deactivated, reset its password, edit profile itself through the application. Direct database editing is needed for those.

The following figure 3.7 use case diagram shows the main activities in Admin module.

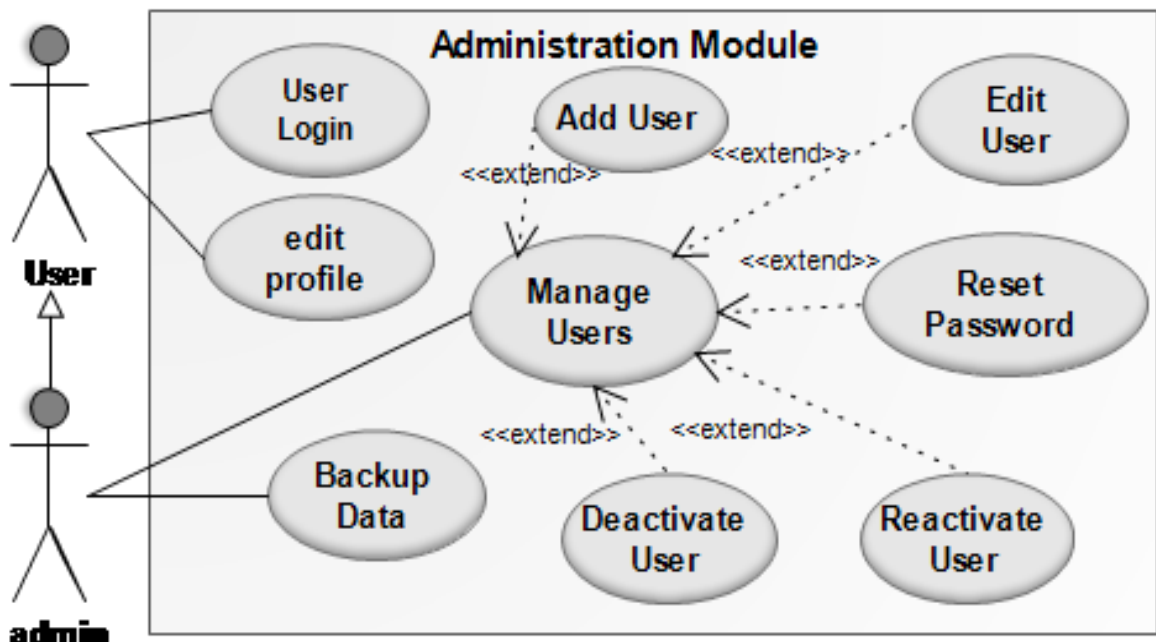


Figure 3.7 Use case diagram for the Administration Module

Use case Name	Manage Users
Actors	Admin
Description	Admin updates users.
Pre-conditions	Admin must logs in to the system
Main Flow	
<ol style="list-style-type: none"> <li>1. Admin navigates to the user administration section of the system.</li> <li>2. System displays the user's details.</li> <li>3. System allows admin to edit, reset password, deactivate or reactivate the user based on the user's current status.</li> <li>4. System updates the user and confirms.</li> </ol>	
Alternate Flows	
<p><b>3.1 Admin selects the option to edit the user.</b> 3.1.1. 'Edit User' use case is executed.</p> <p><b>3.2. Admin selects the option to reset the password of the user.</b> 3.2.1. 'Reset Password' use case is executed.</p> <p><b>3.3. Admin selects the option to deactivate the user.</b> 3.3.1. 'Deactivate User' use case is executed.</p> <p><b>3.4. Admin selects the option to reactivate the user.</b> 3.4.1. 'Reactivate User' use case is executed.</p>	
Post-conditions	none

Table 3.10 Use case narrative for Manage Users use case

The following table 3.11 shows summary of each modules Login privileges.

Modules	Super Admin	Admin	Manager	Operator
Administration	√	√		
Stock	√		√	
Sales	√		√	√
Purchase	√		√	√
Financial	√		√	
Report	√		√	
Home - profile	√	√	√	√

Table 3.11 Access control for modules

The following shows 3.8 sequence diagram of add new user process.

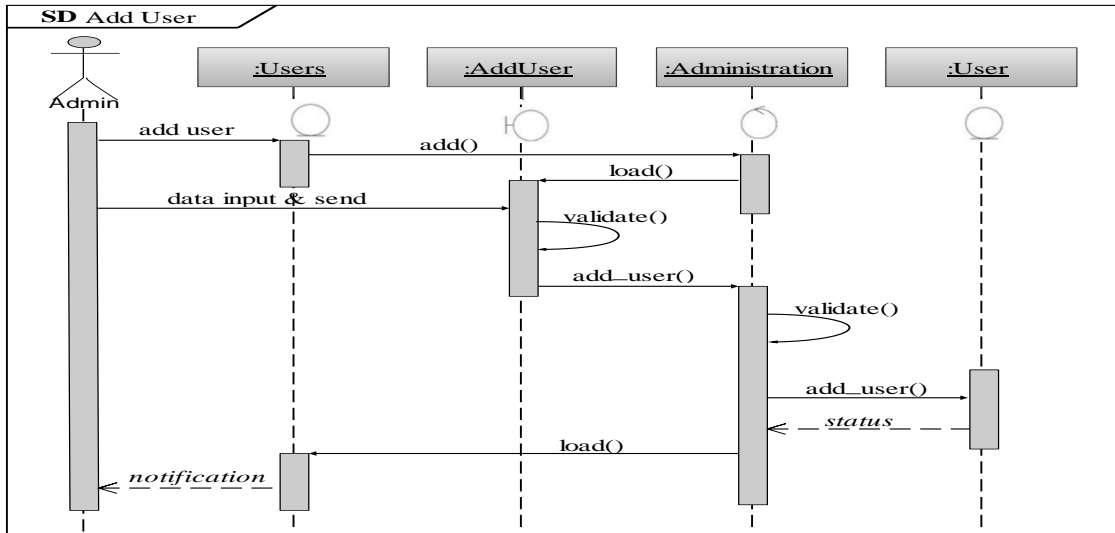


Figure 3.8 Sequence diagram for Add Users use case

### 3.6 Class Diagram of the WBBM

The static view of the application is represented by the class diagram which is very useful to get an overall idea of the system. [11]

The overall class diagram of the WBBM with their multiplicity and the associations is shown in figure 3.9

Please refer the Appendix B - Design Documentation for attributes and methods of each class.

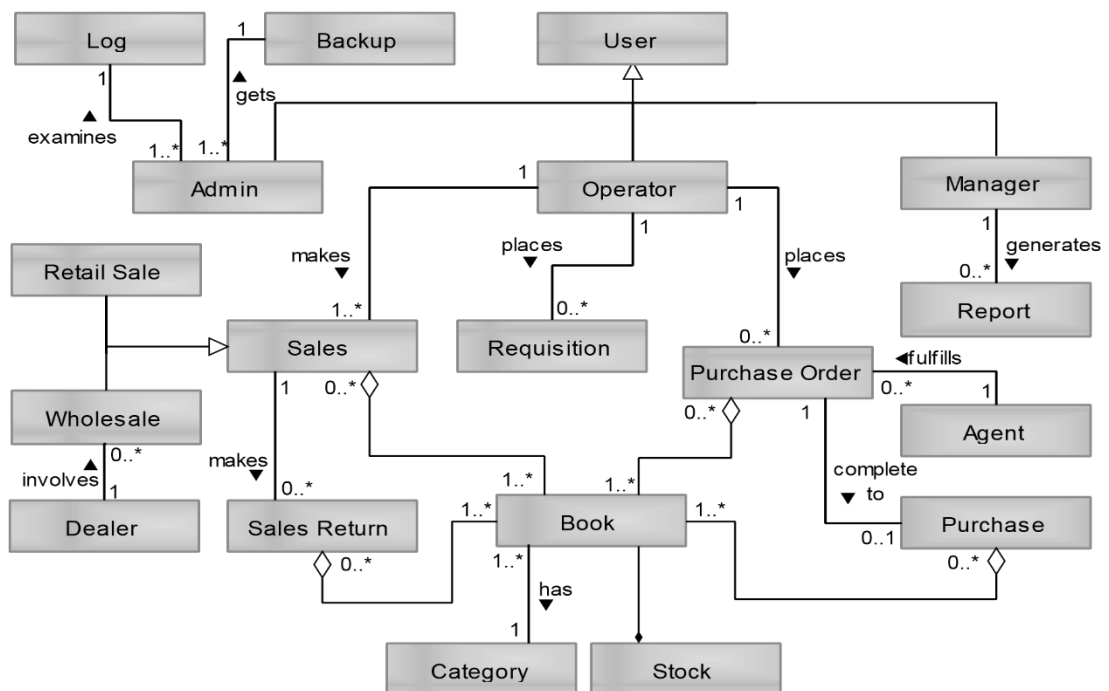


Figure 3.9 Class diagram of the WBBM



### 3.7 Database Design of the WBBM

Database is one of the most critical parts of any data-driven application such as WBBM. Hence appropriate techniques are used in order to ensure the integrity of the database. Normalization is carried out to eliminate possible database anomalies.

Tables are transformed into the first normal form (1NF), by conveying all the multi-value attributes into the single-value attributes. Then the tables are transformed into the second normal form (2NF) and third normal form (3NF) further by eliminating partial functional dependencies and transitive dependencies respectively. [22]

All the tables of the WBBM were transformed into third normal form (3NF).

Following figure 3.10 represents the overall database design of the WBBM.

Please refer the appendix B - Design Documentation for more details of the database design.

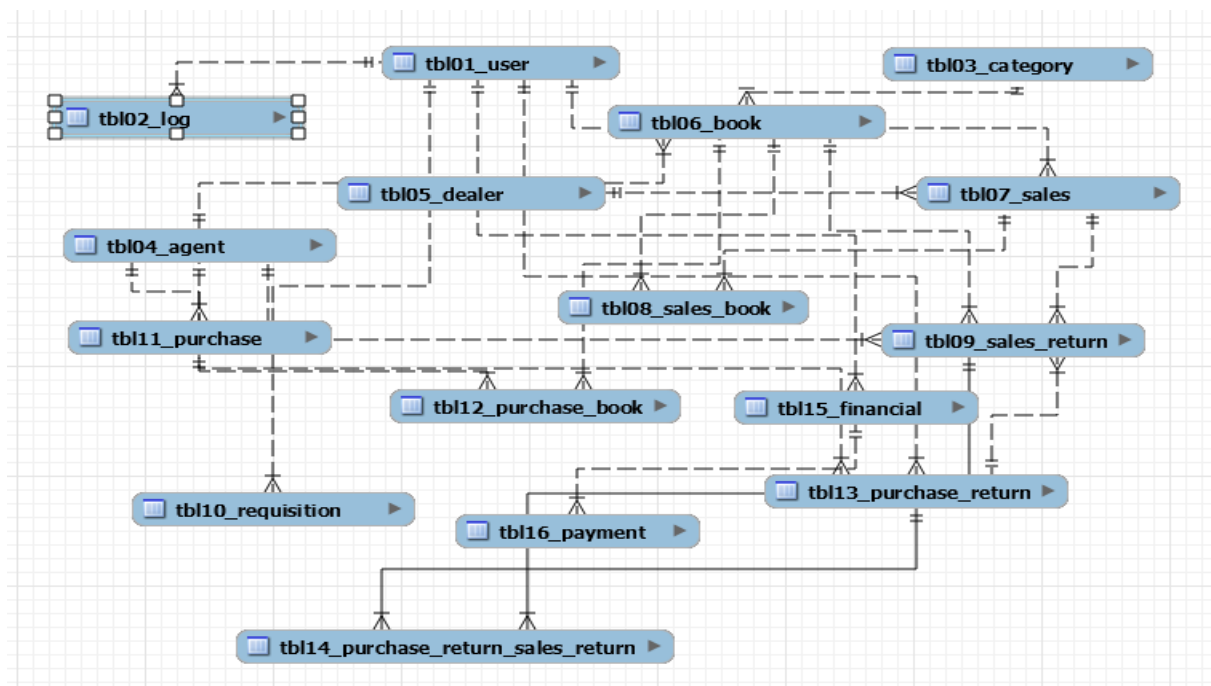


Figure 3.10 Database design of the WBBM

### 3.8 Development Architecture

WBBM is preliminary based on the client-server architecture because of its web based nature. The MVC architecture was selected for the WBBM further which is a software development design pattern and approach which separates the application into three segments: Models, Views, and Controllers. Using this, the system can be structured in

order to promote the reuse of program code, which increases the maintainability and flexibility of the system.

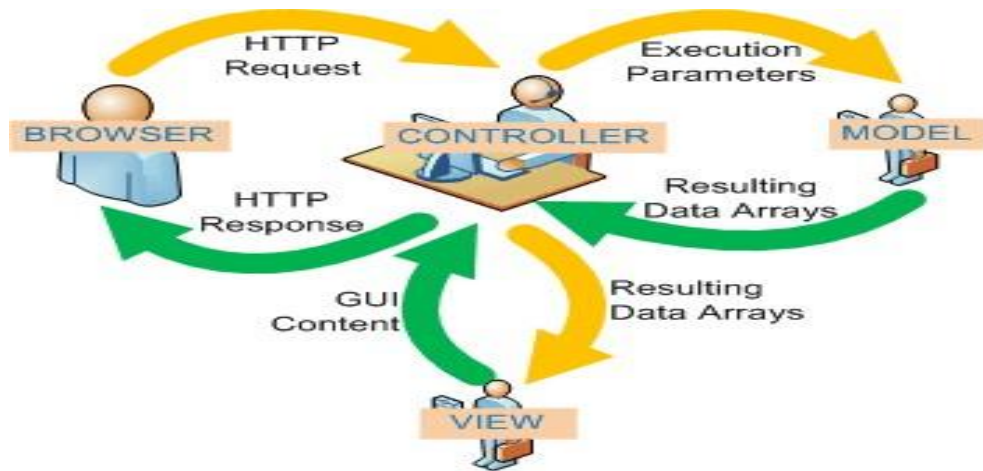


Figure 3.11 MVC Architecture

As shown above in figure 3.11, the entire application is separated into three main components called Model, View and Controller in the MVC architecture. The Controller process the user request. Model is responsible for data manipulation process connected with the database. Views are responsible for representing the output.

### 3.9 User Interface Design

User interfaces are one of the critical factors in evaluating the user friendliness of the application. Because it is the component the user sees and interacts with. Some of the guidelines followed have been summarized below. [21]

- All of the interfaces should have a consistency regarding its characteristics such as colors, text styles, structure of the components and also its behavior such as navigations.
- All the effects and dynamic changes in the interface should provide specific meaning to the user such as links with pointer cursor red color for deletion or deletion warning and so on.
- It should be easy to learn for the users. Interfaces should be as simple as possible and providing good assistance such as tool tips, popup messages and notifications would be promoted this.

- The messages such as error messages and confirmation messages should be consistent, simple and free from technical jargon.
- The data entering process should be improved and captured mistakes as possible for the interfaces such as forms. Providing default values and water marks, using lists and option buttons instead of text boxes for selecting data rather than typing would be helpful.

The following sections represent some of the main interfaces and components of the WBBM in order to provide a perception of the overall interfaces of the WBBM.

Please refer the appendix C - User Documentation for more interfaces.

## Login Interface

The system login page which belongs to the developed system is represents the main interface provided for login to system. In any computerized system the first interface encountered by the user is the login page. Therefore, by designing and handling errors properly a pleasant feeling about the rest of the system can be created within the user.

This facilitates to access the main management areas and provides links for some frequently used functionalities as well.

The following figure 3.12 shows WBBM main login interface.

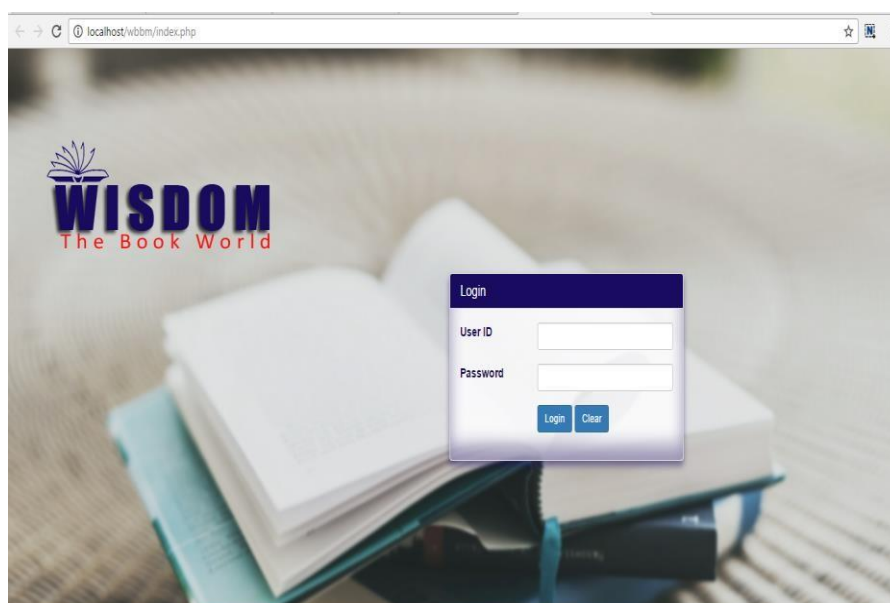


Figure 3.12 Login Interface for the WBBM

## Administration Home Page

The home page of the administrator’s account is displayed by figure 3.13. According to the client’s requirements as well as by studying other similar web-based system interfaces, the final output had been created. The basic information of the user, login time, last login date, and login history is displayed by it. The following figure 3.13 shows Admin page of WBBM.

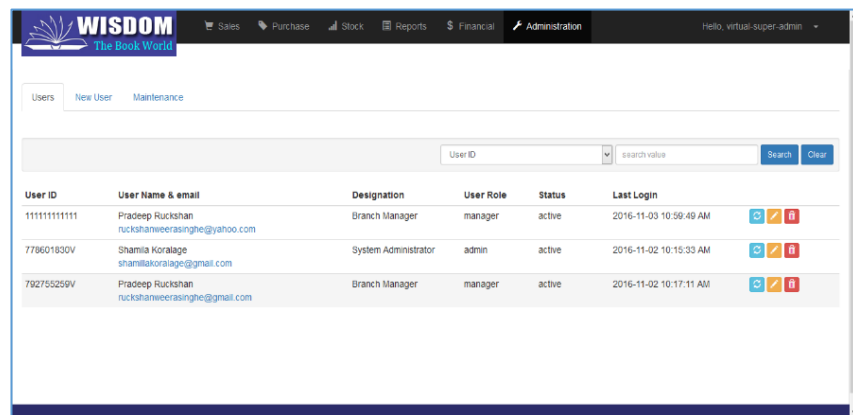


Figure 3.13 Administrator Home page

## Data Grid

Data grid is the data representation structure used throughout the application to represent bulk of data in tabulating format. It also consists of searching & pagination facilities.

The following figure 3.14 shows the data grid used to display products details.

The screenshot shows a data grid interface for displaying payment details. At the top, there is a search bar with a dropdown for 'Payment Method [1 cash, 2 cheque]' and a text input for 'search value', with 'Search' and 'Clear' buttons. The main content is a table with the following data:

Date	Description	Payment Type	Payment Method	Amount
2016-07-23	Sales: 6	income	cash	3,000.00
2016-07-23	Pay credit: Sarasavi Bookshop.	expenditure	cash	6,000.00
2016-07-23	receive debit: lyceum.	income	cash	9,500.00
2016-07-22	Sales: 3	income	cash	5,500.00
2016-07-22	Purchase: 2	expenditure	cash	46,700.00
2016-07-22	Sales: 1	income	cash	9,000.00

Figure 3.14 Data Grid interface of WBBM

## Forms

The below figure 3.15 represents the form used to add new product in to the system. Various form elements have been selected based on the nature of the data to be entered. Error messages are also used to avoid mistakes in the data entering process.

The screenshot shows the WISDOM web application interface. The top navigation bar includes 'Sales', 'Purchase', 'Stock', 'Reports', 'Financial', and 'Administration'. The user is logged in as 'Hello, virtual-super-admin'. The main content area is divided into two sections: 'Categories' and 'Agents'. The 'Categories' section contains a table with the following data:

Category Name	Status
edexcel-maths-ol	Active
general-english	Active
local-grade5	Active
local-maths-ol	Active
edexcel-science-ol	Canceled
test	Canceled

To the right of the 'Categories' table is a 'New Category' form with a text input field labeled '\*category name' and two buttons: 'Add Category' and 'Clear Details'. Below the 'Categories' section is the 'Agents' section, which has a table with columns for 'Agent Name', 'Contact', and 'Status'. To the right of the 'Agents' table is a 'New Agent' form.

Figure 3.15 Form interface of WBBM

# Chapter 4 | Implementation

## 4.1 Introduction

The development of the executable application based on the design which made in the design phase, is brought into action in the implementation phase. Some of the main activities done in this phase are, selecting programming languages, selecting other tools and technologies such as framework, selecting hardware platform and coding the system.

This chapter describes the implementation environment, CodeIgniter framework, development tools used, structure of the application and the explanation of the major code segments.

## 4.2 Implementation Environment

The factors such as cross platform compatibility, distributed web based nature of the application, reliability & performance of the application, design of the WBBM, usage experience, and cost were taken into account when determining the implementation environment of the application.

The implementation environment of the WBBM is given in the following table 4.1.

Hardware Requirement	Software Requirement
Intel Core2Duo processor 2.8GHz	Windows 7 Ultimate 64bit
250GB Hard Disk	WAMP server 2.5 Apache 2.4 PHP 5.5 MySQL 5.6 phpMyAdmin 3.5.1
10 GB Memory per Server	Netbeans 7.1
4 GB RAM	CodeIgniter 2.1.4
Unlimited Monthly bandwidth with Internet	Mozilla Firefox

Table 4.1 Implementation Environment

### 4.2.1 Development Tools

Following tools were used in the development of the WBBM.

- Netbeans 8.1 as the IDE

- Adobe Photoshop CS5 for image editing.
- phpMyAdmin 3.5.1 and MySQL Workbench 5.4 for database manipulation.

#### 4.2.2 Development Technologies

Following technologies were used in the development of the WBBM.

- HTML5 was used to mark-up the web pages. The new elements, attributes and features which have been introduced in the new version are used in order to improve the semantic of the web pages. This is the basic web related language and it helps to keep the system structure clear and conscious.
- CSS3 was used to style the pages. It enables to make consistent design for the whole application in easy-to-manage manner.
- PHP was used as the server sides object oriented scripting language which used when developing the system. [20]
- MySQL was used to implement the database of the application. [22]
- JavaScript was used for the client side scripting. Especially for the client side form validation process. This is a Scripting language and this helps to dynamic the system components.
- JQuery was used to enhance the programming process of the JavaScript and to perform some effects on dynamic web pages. [19]
- AJAX was used to update some part of a web page with some data retrieved from the server, without refreshing the whole page. [18]

#### 4.2.3 Reused Modules

Following modules were used in the WBBM in order to enhance and extend its functionalities. The reliability and compatibility of the modules were considered closely when selecting the suitable modules.

**mPDF** - is a PHP library, which enables to generate pdf files from the given HTML code provides under the GPL license. Some of its major features are easy to use and CSS supportive. [12]

**Jquery** - JQuery is a sub library of JavaScript and is very to use with less code, easy understanding, more features. Rich user interface features and you can work with more than one control or event at a time. JQuery simplifies HTML Document traversing, event handling and Ajax interaction for rapid web development. JQuery is designed to change the way that you write JavaScript. [13]

**PHPMailer** - is a full featured Email Transfer Class for Php. This provides a package of functions to send email. The two primary features are sending HTML Email and Emails with attachments. PHPMailer supports nearly all possibilities to send email. [26] [27]

**Bootstrap** – Bootstrap is a free collection of tools for creating a websites and web applications. It contains HTML and CSS-based design templates. Bootstrap easily and efficiently scales the application with a single code base. Bootstrap makes front-end web development faster and easier. It's made for folks of all skill levels, devices. [14]

**Canvasjs** - JavaScript Charting Library with a simple API and better performance. Charts are responsive & can run across devices. This allows you to create rich dashboards that work across devices without compromising on maintainability or functionality of your web application. Graphs include several good looking themes and are faster than conventional Flash / SVG based Charting Libraries – resulting in lightweight, beautiful and responsive dashboards. [15]

### 4.3 Implementation of CodeIgniter

CodeIgniter (CI) is a free and open source MVC framework based on Php. It is a very powerful framework consists of rich collection of functionalities, which allows developing full featured web applications in efficient manner. It is very simple, easy to use, very flexible and lightweight as well. Because of these features, CodeIgniter has been used as the MVC framework in the implementation of the WBBM. [16]



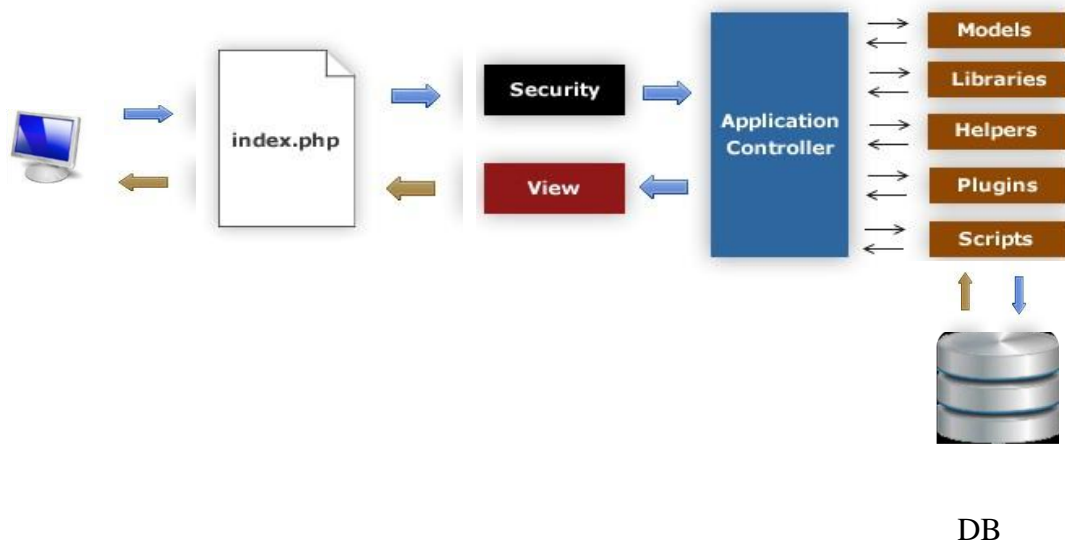


Figure 4.1 Implementation of the CI

As shown above in the figure 4.1, every request is directed to the *index.php* file in which the URL is divided into base URL, controller, method and the set of parameters and then the particular controller is called. Controller processes the request with the assistance of models which provide database manipulation and set of extended functionalities categorized into helpers, libraries and more. Finally, the controller loads the particular view with the data which is sent back to the browser.

#### 4.4 Module Structure

There were some rearrangements made into the design structure of the WBBM when implementing on the CI. Some are splitting some classes into models, views and controllers merging the classes and introducing some additional components such as Helpers and Libraries.

As shown below in figure 4.2, the WBBM consists of two main folders, *application* and *system* which consist of the application specific files and the core CI framework files respectively.

(a) Application Structure.      (b) Structure of the “Models” folder.      (c) Structure of the “Views” folder.

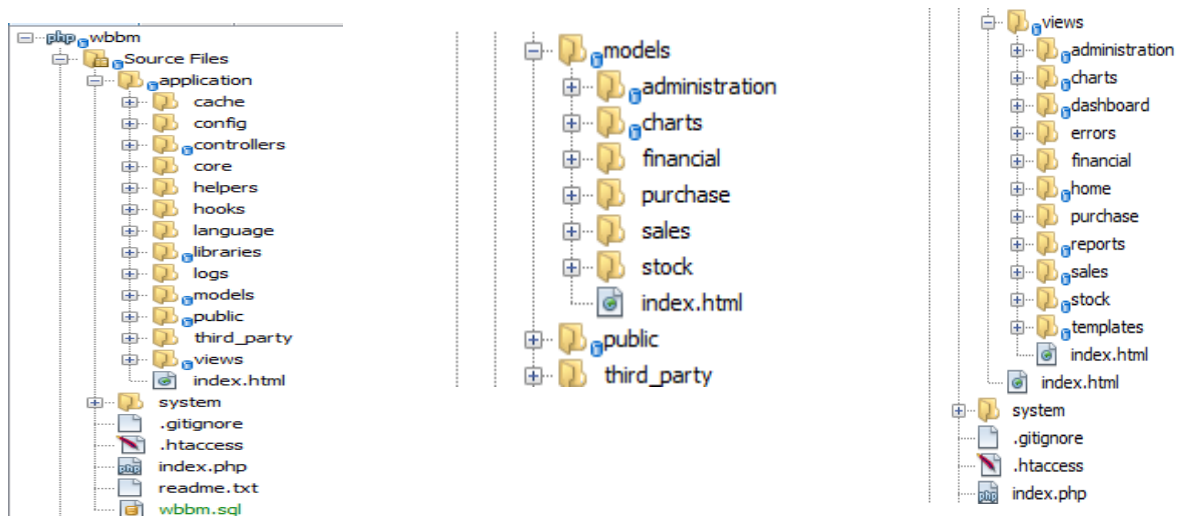


Figure 4.2 Structure of the application

The most important folders in the *application* folder are *controllers*, *models* and *views*. The files in the *models* and *views* folders have been categorized according to the module structure of the WBBM design. The files in the *controllers* are not categorized in this manner to make clean URLs.

*Public* folder consists of the files which are allowed to access publically. It has folders for images, scripts, style sheets and so on.

## 4.5 Major Code Descriptions

Selected code segments have been described in this section in order to provide the clear picture about the execution of the WBBM.

Please refer the appendix F – Code Listings for more codes.

### 4.5.1 Controller

As mentioned earlier, controller is the main component of the MVC framework. Each http request invokes a specific function of a specific controller.

Following code segment shows the overall structure of the *User* section of the *Administration* controller which related to the *user module*.

```
<?php
class Administration extends CI_Controller {
public function __construct() {
public function __construct(){
```

```

public function index($page = 0) {
public function add() {
public function edit($user_id) {
public function add_user() {
public function edit_user($user_id) {
public function deactivate_user($user_id) {
public function reactivate_user($user_id) {
public function reset_password($user_role,$user_id) {
public function user_search() {
public function user_clear_search() {
private function validate_add_user() {
private function validate_edit_user($user_id) {
public function validate_email($str,$user_id) {
public function maintenance($page = 0) {
public function log_search() {
public function log_clear_search() {
public function backup() {
}
?
>

```

Each controller must be extended from *CI\_Controller* class, which is the parent controller of the CI provides core controller functionalities such as loading models. The functions of the controller have been categorized further according to their purpose as commented in the code in order to improve the maintainability of the controller.

The following code segment shows the structure of the *add ()* function, which is responsible for loading the user registration form to add a new user. All main steps of the process have been commented.

```

public function add() {
$this->application->is_login();
$data = $this->application->set_data();
$data["username"] = $this->get_user_name();
$this->load->view("Administration/add_user_view",$data);
}

```

The variable *\$this* represents the object instantiated from the current controller, through which all the functionalities of the controller can be accessed. For an example, following codes load *application* library, and *add\_user\_view* view respectively.

The following code segment shows the important codes of *add\_user()* function, which is responsible for adding the new user based on the data sent by the *add\_user\_view*.

```
public function add_user() {
$this->application->is_login();
$server_validation = $this->
validate_add_user(); if( $server_validation
){
$this->db->trans_start();
$this->load->model("administration/user_model");
$data_set_user = array(
"user_name" => $this->input->post("txt_user_name"),
"role" => "staff"
);
$user_id = $this->user_model->add_user($data_set_user);
$data_set_user = array(
"id" => $user_id,
"user_name" => $this->input->post("txt_user_name"),
"pass_word" => $this->input->post("txt_pass_word"),
);
$this->user_model->add_user($data_set_user);
$this->db->trans_complete();
}
```

Server side data validation is made firstly in order to ensure the accuracy of the data. Further process is begun after passing the validation successfully. Some of the built in Components of the CI, such as *Input class*, *Database class*, *Session library* and *Form validation library* were used in order to get full potential of the framework.

The *post* function of the *Input* class was used to retrieve the data sent through the POST method. The benefit of using this function rather than using *\$\_POST* is that it automatically checks whether the particular variable exists before retrieving the data. Hence it eliminates the additional codes needed to check the existence of the particular variable.

```
$query_status = $this->db->trans_status();
if($query_status) {
$notification = array(
"is_notification" => true,
"notification_type" => "success",
"notification_title" => "Insert User",
"notification_description" => "The new User is added",
"print_url" => "user"
```

```
); }else{
}
$this->session->set_flashdata($notification);
redirect("user");
```

The *set\_flashdata()* function of *Session* library was used to send back the notification information. This function sets data as session data but automatically clears immediately after the next http request. *redirect()* function of the *URL* helper is responsible for navigation to the URL defined as the parameter.

#### 4.5.2 Model

Model is responsible for enabling database interaction. Each model of the WBBM provides a set of functionalities enabling CRUD, which means creating, reading, updating and deleting entries in a table. The models were developed for each table in the database. They have consistent structure and functionality as well.

The following code segment shows the structure of the *user\_model* of the WBBM with some important functions.

```
<?php
class Administration extends CI_Model{
private $table =
"user"; public function
__construct(){
parent::__construct()
; $this->load-
>database();
}
public function get_user($fields="", $criteria="",
$offset="", $limit="", $join1="", $order_by="", $join
2=""){ if( $fields != "" ){
$this->db->select($fields); }
if( $criteria != "" ){
$this->db->where($criteria); }
if($join1!=""){ $this->db-
>join($join1[0], $join1[1], "left"); }
if($join2!=""){ $this->db-
>join($join2[0], $join2[1],
"left"); }
if( $offset != "" ){
$this->db->offset($offset); }
if( $limit != "" ){ $this->db->limit($limit); }
if( $order_by != "" ){ $this->db-
>order_by($order_by); } else{
```

```

$this->db->order_by("user.id", "ASC"); }
return $this->db->get($this->table)-
>result_array();
}
public function add_user($data_set) {
return $this->db->insert($this-
>table, $data_set);
}
public function update_user($data, $criteria) {
return $this->db->update($this->table, $data, $criteria);
}
public function get_total_rows($criteria =
"") {
if( $criteria == "" ){
return $query = $this->db->get($this->table)->num_rows();
} else {
return $query = $this->db-
>get_where( $this-
>table, $criteria)->num_rows();
} } }
?>

```

The functions of *Active Record* class of CI were used for database manipulation rather than using conventional SQL statements directly, because of the benefits such as minimal and more clearly scripting and database independence.

Following example shows the using of some active record functions to retrieve data with the corresponding SQL statement it produces.

```

$fields = array("user_name", "pass_word");
$criteria = array("id" => 1);
$table = "user";
$this->db->select($fields);
$this->db-
>where($criteria);
$this->db-
>get($table); SQL
Statement:
SELECT `user_name`, `pass_word` FROM (`user`) WHERE `id` = 1

```

### 4.5.3 View

Views are the web pages display on the user's browser, which are the combination of HTML and PHP. HTML defines the static contents while PHP for generating dynamic contents on the data submitted by the controller. New HTML5 elements and

attributes have been used in order to enhance the semantics of the web pages.

Following code depicts a part of the view, *add\_user\_view*.

```
<?php
$this->load->view("templates/header");
<title>WBBM|New User</title>
<section class = "form form-margin-left">
<form method = "post"
action = "<?php echo
site_url("user/add_user
");?>"
onsubmit = "return validate_add_user()">
<section class = "form-body">
<div class = "form-subtitle">Admin</div>
<div class = "form-row">
<label class = "form-label">
<span class = "form-data-
required">*</span> User Name</label>
<input type = "text" name =
"txt_user_name" id = "txt_user_name"
value = "<?php echo
set_value("txt_user_name");?>" class =
"form-input" placeholder="user name"/>
<div class = "form-error-text form-
margin" id = "txt_user_name_error">
<?php echo form_error("txt_user_name");?>
</div>
</div>
?>
<?php //footer
$this->load->view("templates/footer");
?>
```

As shown in the above segment, views have been nested to reuse the similar header and footer section for all the pages throughout the application which is more preferable approach rather than using conventional *includes()*, because it eliminates linking errors.

The function *site\_url()* which is a member of built in helper *url*, generates full url to the destination. Another two functions of the *form* helper, *set\_value()* and *set\_error()* preserves the previous value and generates error messages respectively in the event of server side validation failure.

Client side data validation was done using JavaScript in order to save time and bandwidth in the views which allows user to add or edit data. Following code depicts a part of data validation in *add\_user\_view*.

```

function validate_add_user() {
    btn_disable('#btn_add_user');
    var status = true;
    var val ;
    var err ;
    clear_errors();
    val = $('#txt_nic').val();
    err = $('#txt_nic_error');
    if(is_empty(val)) {
        err.text('The NIC No. field is required. ');
        status = false;
    }else if(!is_nic(val)) {
        err.text('The NIC No. field must contain a valid NIC number. ');
        status = false;
    }
    val = $('#txt_first_name').val();
    err = $('#txt_first_name_error');
    if(is_empty(val)) {
        err.text('The first name field is required. ');
        status = false;
    }
    val = $('#txt_last_name').val();
    err = $('#txt_last_name_error');
    if(is_empty(val)) {
        err.text('The last name field is required. ');
        status = false;
    }
    val = $('#txt_email').val();
    err = $('#txt_email_error');
    if(is_empty(val)) {
        err.text('The email field is required. ');
        status = false;
    }else if(!is_email(val)) {
        err.text('The email field must contain a valid email. ');
        status = false;
    }
}

```

As shown the above code, JQuery was used to minimize and simplify the node selecting process. The function *clear\_errors()* clears the all previous errors. The validation functions such as *is\_empty()* have been defined in the external JavaScript file in order to reuse and manage them easily.



#### 4.5.4 Library

Library is a collection of functions which is one type of reusable components in CI such as helpers. CI consists of several built in libraries. It also allows the developer to extend an existing library or creating own library from scratch.

The following code shows the *function level 1()* library which extends the built-in library the purpose of enhancing the server side data validation capability of checking the user levels for access the system with the built-in library by adding more functions for application specific data validations.

```
<?php
public function level1($logout = TRUE) {

    if($this->CI->session->userdata('is_login') && ($this->CI->session->userdata('user_role') == 'super-admin' || $this->CI->session->userdata('user_role') == 'admin'))return TRUE;

    }else{
        if($logout){ redirect('home'); }
        else{ return FALSE; }
    }
}

public function level2($logout = TRUE){

    if($this->CI->session->userdata("is_login") && ($this->CI->session->userdata('user_role') == 'super-admin' || $this->CI->session->userdata('user_role') == 'manager')) return TRUE;

    }else{
        if($logout){ redirect('home'); }
        else{ return FALSE; } } }

public function level3($logout = TRUE) {

    if($this->CI->session->userdata("is_login") && ($this->CI->session->userdata('user_role') == 'super-admin' || $this->CI->session->userdata('user_role') == 'manager' || $this->CI->session->userdata('user_role') == 'operator')){

        return TRUE;
    }else{
        if($logout){ redirect('home'); }
        else{ return FALSE; } } }

public function level4($logout = TRUE) {

    if($this->CI->session->userdata("is_login") && ($this->CI->session->userdata('user_role') == 'admin' || $this->CI->session->userdata('user_role') == 'manager' || $this->CI->session->userdata('user_role') == 'operator')){

        return TRUE;
    }else{
        if($logout){ redirect('home'); }
        else{ return FALSE; }} }?>
```

# Chapter 5 | Evaluation

## 5.1 Introduction

Evaluation is another critical phase of the software development process in which the system is tested against the various characteristics such as functional requirement and non-functional requirement in order to ensure that the system is working properly and meets all of its underlined specifications. [17]

This chapter describes the testing methods, test plan and the test cases, test data and test results and acceptance testing regarding the WBBM.

## 5.2 System Test Plan

Planning a test case is a very important aspect for developing system as well as for the completed system. Test plan should have the ability to test the functionality of the overall system. By properly testing a system, it can identify the errors which generate from the system and can correct them. The implemented system was tested using different test cases.

Since the development started, the test plan continued by testing the system units. After completing a system unit, it was completely tested to identify whether it can function according to expectations. Therefore, early detection of the errors was helped by this testing stage.

After performing the system unit testing, next integration testing was done, and this can identify the errors and the required functionality of the units after integration. System testing was performed as the final stage for the completely developed web based application to check the functionality. This performed at the client's environment with client's original data.

## 5.3 Testing Methods

There are lots of software testing techniques and types have been introduced in software engineering in order to test different aspects of the different systems.

Black Box testing and White Box testing are the key testing techniques used extensively in the software testing process. In Black Box testing the particular function is tested by evaluating the output against the input without considering the

inner structure of the function while in the White Box testing each of the logical paths of the function is tested considering the inner structure of the function.

There were various types of testing used in the different development levels of the WBBM. Unit testing was carried out for each individual functions of each module during the coding phase in order to ensure the correctness of them. Integration testing was carried out after the completion of the unit testing in order to ensure the correctness of the functionality when interacting the modules, each other. System testing was carried out on the whole system after completion of the integrating testing in order to ensure the correctness of the whole system.

## 5.4 Test Plan and Test Cases

Test plan is the documentation which guides the testing process in order to deploy proper software testing. It defines the scope, objectives and the constraints of the testing process, required resources, schedule and especially the test cases.

Test cases were defined as a part of the test plan in which the steps carried out and the expected results are documented for each feature to be tested. Test cases were categorized in to the modules of the WBBM in order to execute the testing process efficiently.

The following sub sections represent the most important test cases defined for each module of the WBBM. Some test cases are common to the whole application, which have been categorized under the “common functions” sub section.

### 5.4.1 Purchase Module

The important test cases for the Purchase module are given in the following table 5.1.

Test No.	Test Description	Steps to Test	Expected Result
1	Add new requisition[valid]	Navigate to sales section and then New Requisition sub section. Fill details and add.	Add the requisition, confirm.
2	View requisitions.	Navigate to requisitions section.	Display the first set of recent requisitions.
3	Edit requisition[valid]	Click Edit Requisition button in particular requisition. Edit not and click Save Changes button.	Requisition updated and confirmed.

4	Cancel Requisition	Click Cancel Requisition button in particular requisition.	Cancel the requisition, confirm.
5	Complete Requisition	Click Complete Requisition button in particular requisition.	Update the status, confirm.
6	Add new Purchase Order	Navigate to purchase section and then New Purchase Order sub section. Fill details and add.	Add the Order, display button to generate purchase order and confirm.
7	Print purchase order	Click Print Purchase Order button in confirmation after adding order.	Generate Purchase Order.
8	View purchases/ purchase orders	Navigate to Purchases section.	Display the first set of recent purchases /purchase orders.
9	View purchase /purchase order	Click View Purchase button in particular entry.	Display the full details of that purchase/ purchase order.
10	Cancel purchase order	Click Cancel Order button in particular purchase order.	Update the purchase order as canceled, confirm.
11	Complete purchase order	Click Complete Order button in particular purchase order. Fill the relevant details and click Complete Order button.	Update the purchase order as completed with other data, confirm.
12	Print GRN	Click Print GRN button in confirmation after completing order.	Generate GRN
13	Add new purchase return order	Navigate New Purchase Return Order sub section. Select returns and add.	Add the Order, display button to generate purchase return order and confirm.
14	Print purchase return order	Click Print Purchase Return Order button in confirmation after adding order.	Generate Purchase Return Order.
15	View purchase return orders	Navigate to Purchase Returns section.	Display the first set of recent purchase returns.
16	View purchase return	Click View Purchase Return button in particular entry.	Display the full details of that purchase return.
17	Cancel purchase return order	Click Cancel Order button in particular purchase return order.	Update the purchase return order as canceled, update related sales return entries & confirm.
18	Complete purchase return order	Click Complete Order button in particular purchase return order.	Update the purchase return order as completed with other data, update stock if needed & confirm.

Table 5.1 Test cases for the Purchase Module

## 5.4.2 Sales Module

The following table 5.2 represents the important test cases for the Sales module.

Test No.	Test Description	Steps to Test	Expected Result
1	View Sales	Navigate to sales section and then sales sub section.	Display the first set of most recent sales.
2	View sale.	Click View Sale button in the particular sale entry.	Display the full details of that sale.
3	Add new Retail Sale[valid]	Go to new Retail Sale, add details and click Add Sale button.	Sale added, confirm and display print invoice button.
4	Add new Whole Sale [valid]	Go to new Whole Sale, add details and click Add Sale button.	Sale added, confirm and display print invoice button.
5	Add new Retail Sale[invalid quantity]	Go to new Retail Sale, add details with a quantity greater than current stock and click Add Sale button.	Display error and re prompt to correct.
6	Add new Retail Sale[existing book]	Go to new Retail Sale, add details with entering similar book twice and click Add Sale button.	Display error and re prompt to correct.
7	Add new Whole Sale [past dealer]	Go to new Whole Sale, examine the dealer list for past dealer.	Past dealer not in the list.
8	Print Invoice	Click Print Invoice button in confirmation after adding sale.	Invoice generated.
9	Search Book	Go to Search Book section, make criteria, click Search button.	Relevant books are listed down.
10	Add new sales return[valid]	Click Add Return button in the particular sales entry, fill details, click Add Return button.	Return added, confirm and display print return notice button.
11	Add new sales return[invalid quantity]	Go to add return, add details with a quantity greater than quantity provided and click Add Return button.	Display error and re prompt to correct.
12	Provide Books for return	Go to add return, add details with a quantity provided and click Add Return button.	Dealer added, confirm and stock updated.
13	Print return notice	Click Print Notice button in confirmation after adding sale.	Generate notice.
14	Complete return	Click Complete Return button in the particular return entry.	Update the return status & stock, confirm.
15	Search Return	Go to Sales Return section, make criteria, click Search button.	The section displays with first set of rows.

Table 5.2 Test cases for the Sales Module

### 5.4.3 Stock Module

The important test cases for the Stock module are given in the following table 5.3.

Test No.	Test Description	Steps to Test	Expected Result
1	View Stock	Navigate to stock section.	Display the first set of books in the stock.
2	Low stock are highlighted.	Check a book in which the current stock is less or equal to minimum stock level.	Highlighted the record in red color.
3	Add new book [valid]	Go to new book, add details and click Add Book button.	Book is added and confirmed.
4	Add new book [invalid/existing]	Go to new book, add details with existing ISBN and click Add Book button.	Display error and re prompt.
5	Edit book[valid]	Click the Edit book button in front of the particular book record, edit relevant fields and confirm.	Update the details and confirm.
6	Edit book[invalid min-stock]	Input number greater than max stock and confirm.	Display error and re prompt to correct.
7	Discontinue book	Click the Discontinue Book button in front of the particular book record and confirm.	Change the book status as inactive & confirm.
8	Discontinue book [non-rollback]	Examine the functionalities provided for inactive book.	No buttons for edit or reactivate.
9	Add new agent[valid]	Go to New Agent section, add details, click Add Agent button.	Agent add and confirm.
10	Add new agent [invalid/existing]	Go to Agent, add details with existing name and click Add Agent button.	Display error and re prompt to correct.
11	Edit agent[valid]	Click the Edit Contact button in front of the particular record, edit contact and confirm.	Update the details and confirm.
12	Add new dealer[valid]	Go to New Dealer section, add details, click Add Dealer button.	Dealer added and confirm.
13	Add new category[valid]	Go to New Category section, add details, click Add Category button.	Category added and confirm.
14	Cancel category	Click the Cancel Category button in front of the particular category.	Change the category status as cancelled & confirm.
15	View Agents, Categories & Dealers	Navigate to Dealers, Categories & Agents section.	The section displays with first set of rows.

Table 5.3 Test cases for the Stock Module

### 5.4.4 Financial Module

Following table 5.4 represents the important test cases for the financial module

Test No.	Test Description	Steps to Test	Expected Result
1	View transactions	Navigate to Financial section.	Display most recent set of transactions.
2	Pay Credit	Select the agent. Fill the amount and click Pay button.	The total balance of credit is decreased, add transaction and confirm.
3	Receive Debit	Select the dealer. Fill the amount and click Receive button.	The total balance of debit is decreased, add transaction and confirm.

Table 5.4 Test cases for the Finance Module

### 5.4.5 Web Module

Following table 5.5 represents the important test cases for the Web module

Test No.	Test Description	Steps to Test	Expected Result
1	Main menu navigation	Click each menu item of the <i>main menu</i> .	Load the relevant page.
2	Place inquiry	Fill in the form in the <i>contact</i> page. Press <i>send</i> button.	Add the inquiry entry & display confirmation
3	Customer/supplier Login.	Enter <i>id</i> and <i>password</i> of the customer/supplier. Click <i>login</i> .	Supplier/customer are able to login.
4	Add item to the shopping cart	Clicks add <i>to the shopping cart</i> button of the particular product.	Update the shopping cart.
5	View shopping cart.	Click <i>view</i> button of the shopping cart area.	Display the shopping cart.
6	Remove item from the shopping cart.	Click <i>remove item</i> button in front of the particular item.	Remove the particular item from the shopping cart.
7	Clear shopping cart.	Click <i>clear</i> button of the shopping cart area.	Remove the entire item in the shopping cart.
8	Place order	Register or login. Click <i>process</i> button.	Add the new order entry, redirected to the <i>orders</i> page & display confirmation.
9	View order history	Login to the customer account. Click <i>orders</i> menu item.	Display the orders placed by the customer
10	Cancel order	Click <i>cancel order</i> button in	Cancel the order & display

		front of the particular order.	confirmation.
11	Get customer quotation	Click <i>customer quotation</i> button in front of the particular order.	Generate the customer quotation
12	Edit profile	Edit the profile in <i>profile</i> page. Click <i>save changes</i> button.	Save the changes & display confirmation
13	Change password	Enter similar password for <i>new password &amp; retype password</i> fields. Click <i>save changes</i> button.	Save the new password & display confirmation

Table 5.5 Test cases for the Web Module

#### 5.4. 6 Report Module

Following table 5.6 represents the important test cases for the Report module.

Test No.	Test Description	Steps to Test	Expected Result
1	Generate monthly sales report	Navigate to Reports section, select year, month and dealer. click Generate Sales Report button.	Monthly Sales Report is Generated.
2	Generate annual sales report	Select year and dealer. click Generate Sales Report button.	Annual Sales Report is Generated.
3	Generate monthly purchase report	Select year, month and agent. click Generate Purchase Report button.	Monthly Purchase Report is Generated.
4	Generate annual purchase report	Select year and agent. click Generate Purchase Report button.	Annual Purchase Report is Generated.
5	Generate Stock report	Navigate to Reports section, click Generate Stock Report button.	Stock Report is Generated.

Table 5.6 Test cases for the Report Module

#### 5.4.7 Administration Module

Following table 5.7 represents the important test cases for the Administration module.

Test No.	Test Description	Steps to Test	Expected Result
1	User Login[valid]	Enter valid user ID and password. Click login button.	Load the relevant default section according to the user role.
2	User Login [invalid]	Enter invalid user ID and/or password. Click login button.	Display error & redirect to the home page.
3	User Login [invalid/empty]	Click login button without entering user ID and/or password.	Display error & redirect to the home page.
4	Display Greeting	Login and check top right corner.	Should consist user first



	section.		name and user role.
5	Admin cannot be deactivated.	Check admin entry in user's section.	No button to deactivate admin.
6	Admin cannot be edited.	Check admin entry in user's section.	No button to edit admin.
7	Deactivate User	Click deactivate button in user entry.	Change the user status as inactive & display confirmation.
8	Reactivate User	Click reactivate button in user entry.	Change the user status as active & display confirmation.
9	Reset user password	Click reset password button in user.	Change user password to default, confirm.
10	Edit User	Click edit user button, make changes and click save button.	Save changes, confirm.
11	Input NIC [valid]	go to New User, enter NIC with correct format old/new, click Add User button.	No any error
12	Input NIC [invalid/existing]	go to New User, enter an existing NIC, click Add User button.	Display error and prompt to correct.
13	Search Log	go to maintenance, select and enter User ID, click Search button.	Display entries relevant to that user.
14	Data Backup	Click Generate Backup button	Generate backup file.
15	Edit user profile[valid]	Click My Profile in top right corner menu, enter new password, confirm	Save the changes and confirm.

Table 5.7 Test cases for the Admin Module

## 5.5 Test Data and Test Results

The collection of data which is used in the testing process is referred to as test data. Some of the guidelines were considered closely when preparing test data, such as selecting not only valid datasets but also invalid datasets to cover all the aspects of testing process and selecting minimal data set needed to avoid unnecessary complexity.

Dummy datasets were used in the unit testing and the integration testing while the datasets extracted from the existing system were used in the system testing of WBBM.

The test results generated in the execution of the test cases were also documented which can be very useful in the further evaluation and the future maintenance processes.

Please refer the appendix E - Test Results for the test results of the WBBM.

## 5.6 Acceptance Testing

After completing the system testing the system was exposed to its real users to examine the system at the client's premises with own data. The users were selected as representing the all business process of the WB. Some of the customers and the suppliers who are close with the client were also invited to assess the functionalities of the frontend website for the evaluation.

Structured questionnaire was used to get user feedbacks under the set of criteria and to get further suggestions. Then all the feedbacks were summarized and assessed while the minor modifications were also implemented according to the suggestions of the users.

User Acceptance Testing was started by feeding the actual dataset to the developed system. There were some minor modifications required for the system.

The overall system was tested by the client; changing the user's privileges. Because at the requirement gathering stage, it was stated there is a need for having all the company internal operational privileges to Administrator account. After testing the system, it had been requested to test the system using the staff members. Manager, Accountant and Operator had tested the system by login to their user accounts under the direction. Some minor modifications needed were pointed out by the Accountant. When finishing their session there was a pleasant response about the system. According to the business knowledge of the client, as well as the wholesale dealer's account was tested. Those users who deal outside the company premises; had already informed the client about the criteria they should test in the system according to their requirements.

After completing the user acceptance testing, a positive response was received from all the users requesting for minor modifications. With the newly developed system the company could function efficiently and smoothly rather than continue using the old method, the client stated.

Please refer the appendix E - Test Results for the assessment of user feed backs.

The client certificate has been appended to the appendix G as well.

# Chapter 6 | Conclusion

## 6.1 Introduction

It is very crucial for any project to review the works that done throughout the project regarding the challenges faced, mistakes happened and the way corrected them and the knowledge and experience gained.

WBBM was developed as the solution for the problems faced by the WB in achieving their business objectives. The existing system was examined and requirements were gathered in the analysis phase. Then the solution was designed especially using object oriented design techniques. The system was developed in the implementation phase as a web application and finally it was evaluated and ensured that it has met all its objectives successfully.

The rest of the sections describe critical evaluation of the system, future enhancements and the lessons learnt.

## 6.2 Critical Assessment of the Project

Although the system was evaluated as all its objectives were achieved successfully, it is still worthy to self-assess the system from the developer's point of view.

Purchase, sales, stock and administration modules of the backend application were identified as the most critical pieces of the system by considering the objectives and hence paid more attention for them in the development process.

However, because of some unmanageable external issues such as browser supportive, mature of the technology and the expense, it was unable to use some enhanced features for the current system further.

Most of them have been suggested for the future improvement of the system and would be increased the potential of the system in the future.

## 6.3 Future Improvements

Although the WBBM was full filled all of its requirements successfully some of the improvements have been identified throughout the development process which would be considered in the future versions of the WBBM.

- Introducing Barcode System.

Add more effective Bar Cord Reader support operational level making more effective of data capturing. Data entering accuracy of the sales and purchasing activities can be improved using system generated barcodes for the items for their SKUs.

- Integrated application and back office functions related to technology of current website and internal system as Enterprise Resource Planning (ERP) to manage the business in future.
- Provide SMS facility after processing orders and for special tasks by implementing this suggestion, users can get relevant information from the system without accessing the system or email account.
- Improve the Security by placing a standard Firewall, Gateways and use Cryptographic methods to constrict further security as the Intranet deals with large amount of Sensitive Information.
- Obtain a SSL Certificate to conduct all the transactions through a Secure Channel with https.
- Plan to obtain an Intellectual Properties (IPR) for Company Website.

## 6.4 Lessons Learnt

As a student taking a degree program, this was a great opportunity for me to apply the previously learned lessons into a working system. The domain area was considerably large, therefore it helped to gain experience in many business techniques and it broadened my horizons into understanding, how to map those related business processes into a computerized system. It also provides me a valuable opportunity to practice the knowledge and the skill gained during the BIT program me.

The development of the WBBM helps to gain a solid experience of developing a system going throughout the SDLC. The analysis and designing phases provide opportunity to practice various software engineering approaches such as requirement gathering techniques, OOD techniques and designing systems using UML which would be a real benefit in my future career. Using CodeIgniter helps me to gain a solid experience on MVC in the implementation. This phase also helps me to improve my knowledge on Php, AJAX, JQuery and so on. Evaluation phase was a good platform for me to practice how to evaluate a system especially in the client side.

Better experience was obtained on how to communicate with a real customer.

Furthermore, the working environment consisted with web related languages.

## REFERENCES

- [1] searchsoftwarequality.techtarget.com - Search Software Quality. [Online]. Available: <http://searchsoftwarequality.techtarget.com/definition/requirements-analysis/> [Accessed: 10 - Jan- 2016].
- [2] Wikipedia.org-Wikipedia Website [Online]. Available: [http://en.wikipedia.org/wiki/Functional\\_requirement](http://en.wikipedia.org/wiki/Functional_requirement) [Accessed: 01 - Mar- 2016].
- [3] Wikipedia.org- Wikipedia Website [Online]. Available: [http://en.wikipedia.org/wiki/Non-functional\\_requirement](http://en.wikipedia.org/wiki/Non-functional_requirement) [Accessed: 14 - Mar- 2016]
- [4] www.salesbinder.com-SalesBinder Website [Online]. Available: <https://www.salesbinder.com/tour/orders-billing-invoices-estimates/> [Accessed: 24 - Apr- 2016]
- [5] www.vijithayapa.com - Vijitha yapa Website [Online]. Available: <http://www.vijithayapa.com/product/browse/> [Accessed: 04 - May- 2016]
- [6] www.sarasagroup.net-Sarasa Softsolutions Website [Online]. Available: <http://www.sarasagroup.net/products.html/Sales/Inventory/Management/Systems> [Accessed: 04 - Jun- 2016]
- [7] www.retailit.lk - Retail IT Website [Online]. Available: <http://www.retailit.lk/peripherals/> [Accessed: 12 - Mar- 2016]
- [8] www.quickbooks.intuit.com-Quickbooks web site [Online]. Available: [https://quickbooks.intuit.com/inventory\\_conrol/](https://quickbooks.intuit.com/inventory_conrol/) [Accessed: 23 - Mar- 2016]
- [9] www.wikipedia.org - RUP Website [Online]. Available: <http://en.wikipedia.org/wiki/File:Development-iterative.gif> [Accessed: 21 - Apr- 2016]
- [10] www.wikipedia.org - Wikipedia Website [Online]. Available: [http://en.wikipedia.org/wiki/Object-oriented\\_analysis\\_and\\_design](http://en.wikipedia.org/wiki/Object-oriented_analysis_and_design) [Accessed: 28 - Mar- 2016]
- [11] www.tutorialspoint.com - Tutorialspoint Website [Online]. Available: [http://www.tutorialspoint.com/uml/uml\\_class\\_diagram.htm](http://www.tutorialspoint.com/uml/uml_class_diagram.htm) [Accessed: 08 - Apr- 2016]
- [12] www.mpdf1.com-mPDF, Website [Online]. Available: <http://www.mpdf1.com/mpdf/index.php> [Accessed: 18 - Jun- 2016]

- [13] [www.w3schools.com](http://www.w3schools.com) - JQuery UI, Website [Online]. Available: [http://www.w3schools.com/jquery/jquery\\_slide.asp](http://www.w3schools.com/jquery/jquery_slide.asp) [Accessed: 15 - May- 2016]
- [14] [www.wrapbootstrap.com](http://www.wrapbootstrap.com)-Bootstrap Website [Online]. Available: <https://wrapbootstrap.com/themes> [Accessed: 05 - May- 2016]
- [15] [www.cdnjs.com](http://www.cdnjs.com)-Canvasjs Website [Online]. Available: <https://cdnjs.com/libraries/canvasjs> [Accessed: 15 - Jul- 2016]
- [16] [Www. ellislab.com](http://www.ellislab.com) - CodeIgniter Website [Online]. Available: CodeIgniter/EllisLab, <http://ellislab.com/codeigniter> [Accessed: 18 - Jan- 2016]
- [17] [Www. wikipedia.org/](http://www.wikipedia.org/) - Wikipedia Website [Online]. Available: <http://en.wikipedia.org/wiki/Evaluation> [Accessed: 04 - Apr- 2016]
- [18] [Www. w3schools.com](http://www.w3schools.com) - W3Schools Website [Online]. Available: <http://www.w3schools.com/ajax/> [Accessed: 20 - May- 2016]
- [19] [www. jqueryui.com](http://www.jqueryui.com) - JQuery UI, [Online]. Available: <http://jqueryui.com> [Accessed: 15 - May- 2016]
- [20] [www. php.net](http://www.php.net) - Php Website [Online]. Available: <http://php.net/manual/en/function.json-encode.php> [Accessed: 15 - Jan- 2016]
- [21] I. Sommerville, *Software Engineering*, 8th ed. Addison-Wesley, 2007.
- [22] Thomas Connolly & Carolyn Begg, *Database Systems - A Practical Approach to Design Implementation and Management*, 4th edition, Pearson Education, 2005.
- [23] Per Kroll & Philippe Kruchten, *Rational Unified Process Made Easy: A Practitioner's Guide to the RUP*, 1st edition, Addison Wesley, 2003
- [24] Jeffery L Whitten & Lonnie D Bently, *System Analysis and Design Methods*, 7th edition, Tata McGraw Hill, 2007
- [25] J.L. Whitten and L.D. Bentley, *Systems Analysis and Design Methods*, 7th ed. Tata McGraw-Hill, 2007.
- [26] [www.phpmailer.com](http://www.phpmailer.com) - PHPMailer, PHP mailing library. [Online]. Available: <http://phpmailer.worxware.com> [Accessed: 22 - Aug- 2016]
- [27] [www.110mb.com](http://www.110mb.com) - HOWTO: Send emails using PHPMailer and Gmail [Online]. Available: <http://www.110mb.com/forum/index.php?topic=14144.0> [Accessed: 30 - Aug- 2016]

# Appendix A | System Documentation

This documentation provides the details and guidance needed to install and configure the WBBM on a local computer using wamp.

These steps explain about the hardware and software environment which needs to be installed. When installing the system, this documentation can be followed by the interested parties.

## System Requirements

Following table shows hardware and software requirements are recommended for the system on which the WBBM to install, given in Tables A.1 and A.2.

Hardware	Recommended requirement
<b>Processor</b>	2.8GHz Intel Core2Duo or higher
<b>RAM</b>	1GB or more
<b>Hard Disk Space</b>	2GB or more
<b>Display</b>	1366 x 768 resolution
<b>Printer</b>	Inkjet or Laser Printer
<b>Internet</b>	512kbps ADSL connection

Table A.1 Recommended Hardware Requirements

Software	Recommended requirement
<b>Operating System</b>	Windows Xp, Windows 7, Windows 8 O/S
<b>Software bundle</b>	WAMP 2.2 (compatible LAMP or XAMPP) Apache 2.2.22, PHP 5.4.3, MySQL 5.5.24
<b>Browser</b>	Firefox, Chrome
<b>Code Editor</b>	Notepad++ 6.9, Netbeans 8.1
<b>Image Editor</b>	Adobe Photoshop CS5
<b>PDF Editor</b>	MPDF

Table A.2 Recommended Software Requirements

## Application Installation

### 1<sup>st</sup> Step - Installing WAMP and Web browser

To install into the local computer,

Copy the *wbbm* folder in the *software* folder of the CD in to the *www* folder of the WAMP server.

The path of the **www** folder would be *C:/programfiles/wamp/www*.

Note that the path can be changed according to the location where the WAMP server installed.

### Installing Web Browsers

- Install Browsers (refer Table A.2 for the Version and Recommended Browsers).

#### 2<sup>nd</sup> Step - Files Extraction

- Open the CD and copy the WBBM folder and paste it to the directory path  
“C:/wamp/www/wbbm”

#### 3<sup>rd</sup> Step - Database Installation

This section guides you how to install the database using *phpMyAdmin*. Please refer the vendor’s documentations if you use another tool for database installation.

1. To install on the local computer,

- Run the WAMP server
- Open your browser, and type the URL,  
<http://localhost/phpmyadmin/>

and enter Username and Password (if you set username and password).

2. Create empty database by providing name as “**wbbm**”
3. Navigate the *Import* tab and click “**choose file**” button then *Browse* button.
4. Select the file *wbbm.sql* file by opening the folder *Database* on the CD. 5. Then Press “**GO**” button to install located in the bottom of the page.

#### 4<sup>th</sup> step - Database Configuration

The database is configured to the default configuration of the MySQL. Therefore, it is no need to make special configuration unless you use either remote server or login with particular user name and password instead of default settings.

1. Open the file *database.php* along the path *application/config*
2. Specify the server address, user name and the password under the *hostname*, *username* and *password* respectively. Please refer the service provider’s documentation to find these details in the case of using remote server.
3. Save the file and close.



### 5<sup>th</sup> step - Application Configuration

There is a set of parameters available for you to configure the application such as default password, default email address and so on.

1. Open the file *config.php* along the path *application/config*.
2. Change the parameters you need under the *Application Settings*. Refer the comments to identify the purpose of each parameter.
3. Save the file and Close.

### 6<sup>th</sup> step - (Launching System)

1. Verify the WAMP is running; go to the “C:\wampp” Use following ways to access the frontend homepage. Open the installed web browser and type the URL
  - For the local computer: *http://localhost/wbbm* or *http://127.0.0.1/wbbm*
2. Press “Enter” button to access the system.  
Access the admin panel using the user ID and password.
3. Then find the user id of the Super Admin account under the user name *sa* to access the backend application with default password. (Super Admin is the start-up account provided you to log to the backend & create the actual users).

It is strongly recommended you to change admin password immediately after the first login in order to ensure the security of the application. Also you should deactivate Super Admin account after creating the manager, account and should not be used for any other thing.

## Appendix B | Design Documentation

### Use case Diagram with detailed information

Following is the use case diagram in detail shown in Chapter 3. To understand it easily and clearly, it was categorized into six modules and each module explained with a small description. It is important to say that every module consists of a relevant module report. The following tables and figures provide additional use case narratives and design diagrams such as activity diagrams and sequence diagrams related to the section 3.5.

### Purchase Module

The ordered books are marked as Ordered in the stock when placing the order. Hence not allowed to be ordered again until the order is completed or cancelled. The only the books have been registered in the system are allowed to be ordered, means you need to add the book first before if it is a non-existing/ new book.

The purchases of which the status is “Ordered” are purchase orders. The status of new purchase/purchase order defined as “**Ordered**” rather than the most common “**Pending**”, the reason is this is real time process, so placing order means we ordered it. So the word “Ordered” is more suitable. The similar case in purchase return. But when placing sales return, it will be “**pending**”, because at the time we add it we don’t send it for the agent, we do it after sometimes as a bulk.

**List amount** =  $\Sigma$  (Ordered price X Quantity Purchased)

**Purchased amount** =  $\Sigma$  (purchased price X Quantity Purchased) – bill discount

Use case Name	View Purchase
Actors	Operator, Manager
Description	Actor views a purchase in detail.
Pre-conditions	Actor must log in to the system.
Main Flow	
<ol style="list-style-type: none"> <li>1. Actor navigates to ‘Purchase’ section.</li> <li>2. System displays the section.</li> <li>3. Actor clicks the ‘View Purchase’ button in front of the particular purchase.</li> <li>4. System displays the details of the purchase.</li> </ol>	
Alternate Flows	

none	
Post-conditions	None

Table B.1 Use case narrative for View Purchase

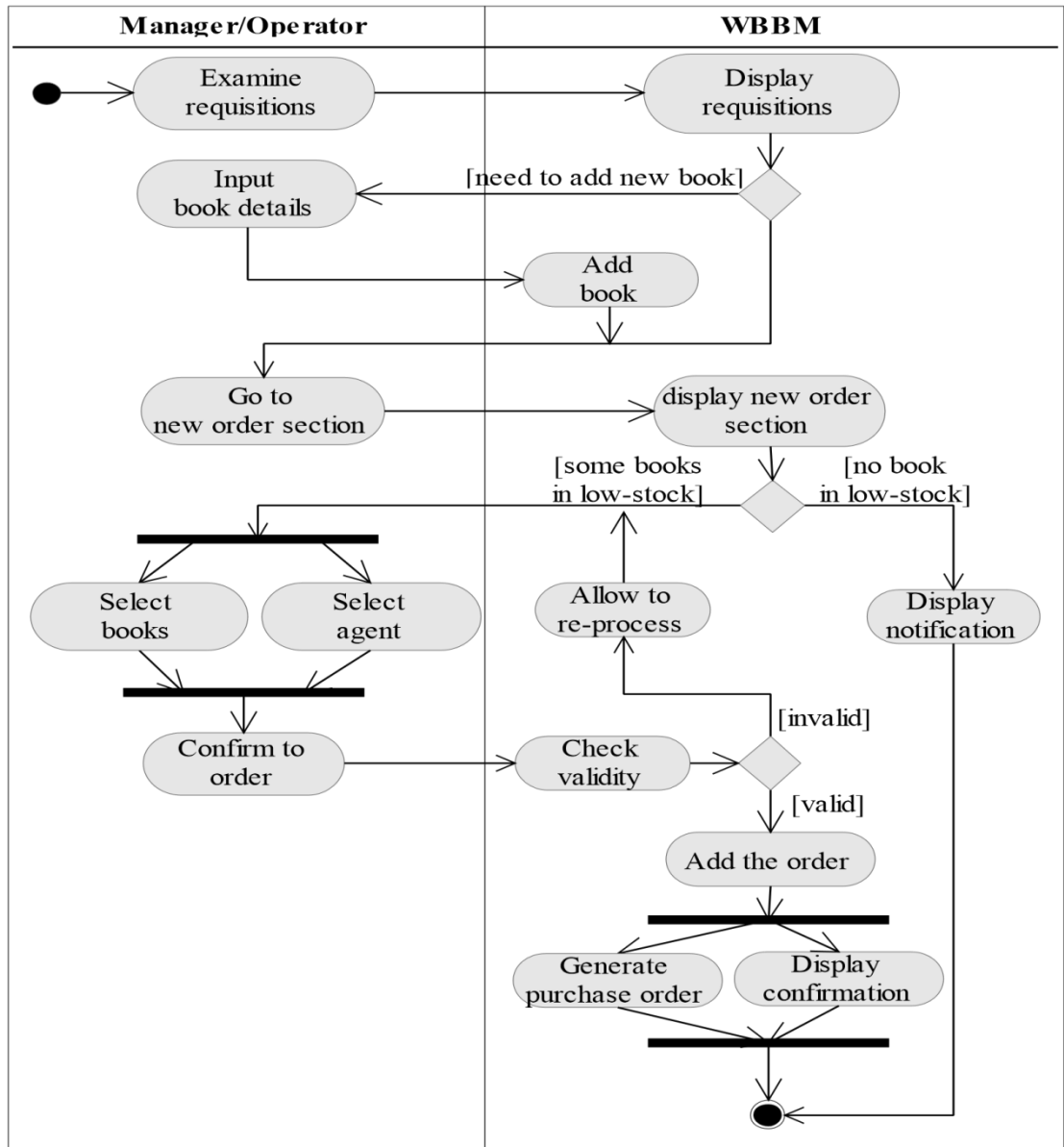


Figure B1 Activity diagram for Purchase Module

## Sales Module

Sales return can be added for a particular sale in which, It must be return within ten date form date sold for retail sales. It must be returned within three months from the date sold for wholesales. The payment history not shown in retail, because the payment paid once at the moment sales made in cash, hence no payment history. It is

shown in wholesale type. The dealer name displayed only in wholesales. The total amount changed when a book is added or removed and the net amount set to the zero. The bill discount is the dealer discount defined for the particular dealer for wholesale. When calculating Maximum Allowable Quantity to be returned for “pending” and “received” sales returns related to the particular sale. The received ones are also taken in to the account, because they are still in the store, not picked by the customer yet should specify the quantity returned, quantity given at the moment. Discontinued books, past agents are also possible to return, but shows in red color. This is because it has not any formal way to decide the return ability based on these things. So only such notifications displayed to make the decision easily. A past dealer also possible to make sales return. Some of the calculations use in Sales Module.

**List Amount** = List Price X Quantity

**Amount Sold** = Price Sold\* X Quantity (\*Price Sold = List Price – Book Discount)

**Total Amount** =  $\Sigma \{(list\ price - book\ discount) \times Quantity\ Purchased\}$

**Net Amount** = Total Amount - bill Discount \*

**Quantity Sold** -  $\Sigma (quantity\ returned - quantity\ given\ on\ time)$

Use case Name	View Sale
Actors	none
Description	Actor views a particular sale.
Pre-conditions	The use case “Manage Sales” must have been executed.
Main Flow	
<ol style="list-style-type: none"> <li>1. Actor clicks ‘View Sale’ button in the particular sales entry.</li> <li>2. System displays the sale in detail.</li> </ol>	
Alternate Flows	
none	
Post-conditions	none

Table B.2 Use case narrative for View Sales

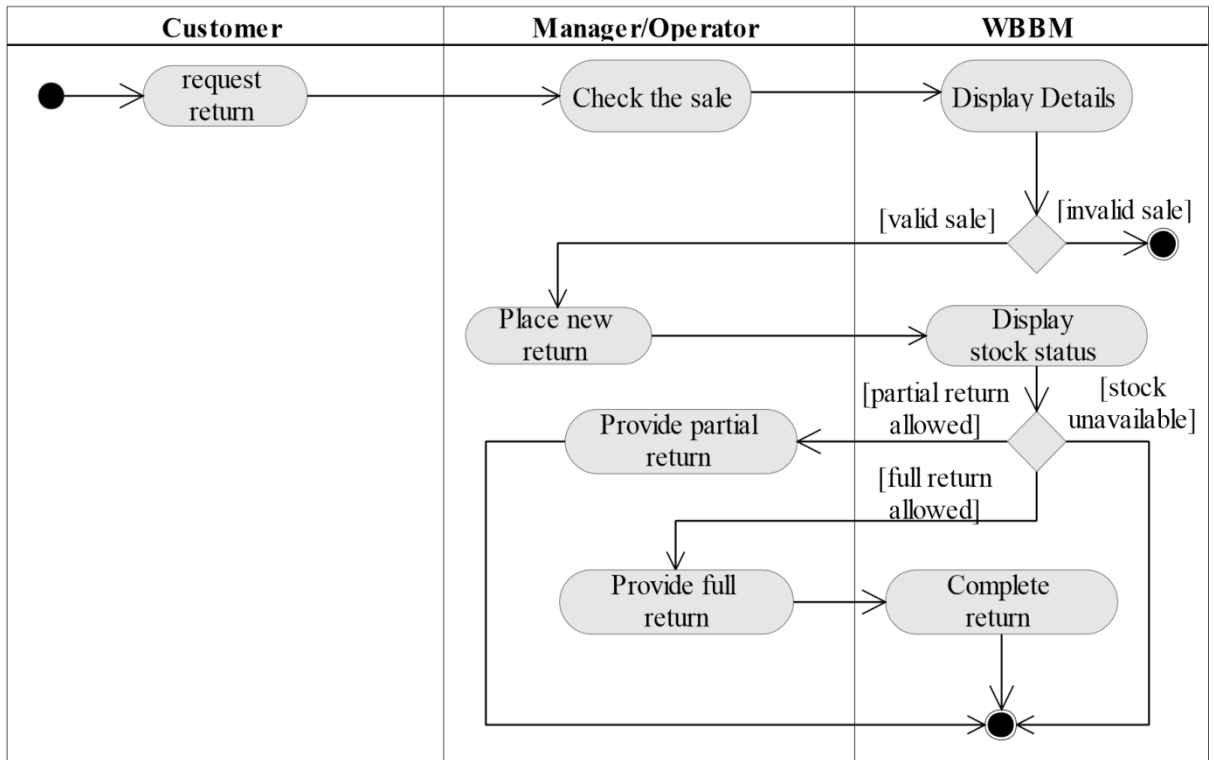


Figure B.2 Activity Diagram for Sales Module

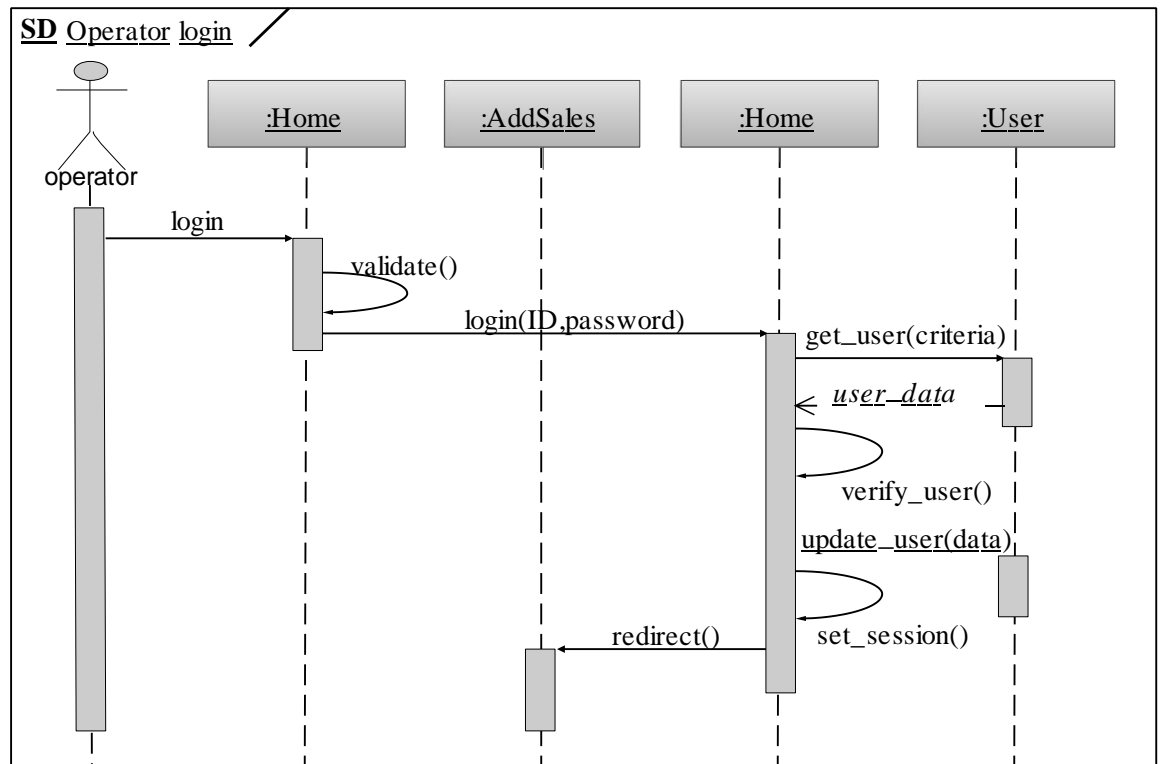


Figure B.3 Sequence Diagram for Add SalesModule

## Stock Module

Some of the parameters of the book are editable, such as category, prices and so on. However, the purchase price is update automatically when a purchase related to that book is completed. Discontinue books are to avoid ordering an out-of-print book. So selling, returning is available for discontinued books as well. Discontinued book means out-of-print, so cannot be reactivated.

The Books are representing the stock and books details itself. Book is related to only one category. The book ISBN and agent combination is unique. This allows you to assign many agents for one ISBN. Which the current agent is canceled and need to purchase from other agent. In that case you need to add it as a new book with similar ISBN and different agent. The books which have less stock than min stock highlighted in red. The discontinued categories and agents are shown in red color.

**Amount listed** =  $\Sigma$  (list price X quantity sold)

**Amount sold** – last sold price that is,

$\Sigma$  {list price – book discount) X quantity sold} – bill discount

Use case Name	Manage Dealers
Actors	Manager
Description	Manager manages dealers' details.
Pre-conditions	Manager must log in to the system.
Main Flow	
<ol style="list-style-type: none"> <li>1. Manager navigates to 'dealers, agents and categories' section.</li> <li>2. System displays the section with options for edit, cancel and reactivate the dealer.</li> <li>3. System saves the changes and confirms.</li> </ol>	
Alternate Flows	
<p><b>1.1 Manager adds a new dealer.</b></p> <ol style="list-style-type: none"> <li>1.1.1 Manager navigates to 'New dealer, agent and category' section.</li> <li>1.1.2 System displays the section</li> <li>1.1.3 Manager fills 'New Dealer' form and confirms.</li> <li>1.1.4 System validates the inputs.</li> </ol> <p><b>2.1 Manager edits a dealer.</b></p> <ol style="list-style-type: none"> <li>2.1.1 Manager clicks 'Edit Dealer' button.</li> <li>2.1.2 System popups the form to edit.</li> <li>2.1.3 Manager Edits and save changes.</li> <li>2.1.4 System validates the inputs.</li> </ol> <p><b>2.2 Manager cancels a dealer.</b></p> <ol style="list-style-type: none"> <li>2.2.1 Manager clicks 'Cancel Dealer' button.</li> </ol> <p><b>2.3 Manager reactivates a dealer.</b></p>	

2.3.1 Manager clicks ‘Reactivate Dealer’ button.	
Post-conditions	The dealers managed successfully.

Table B.3 Use case narrative for Manage Dealers

### Financial Module

The total credit limit keeps with the agents table as a field, to get directly rather than calculating as above. When process, Agents credit limit decreased, purchases which have credits, selected from the earlier to latest one and reduce the credit until the total paid amount is over. Receive debit similar to credit, the sales related to the particular dealer, in which  $\Sigma$  (Net Amount – Total Paid) are selected. When process is similar, that is, zero (o) reduces the debit amount sales which have debits, selected from the earlier to latest one and reduce the debit until the total paid amount is over.

View transactions type can be either sales (direct sales), credit (purchase) or debit (wholesales) view payment credit and debit type transactions has view option to view the sales / purchase involved with the amount paid. But sales are always single-entity payment involved in the sales itself mentioned in the description, hence no payment history to be displayed. Complete transaction debit payment and credit payments made in cheque are allowed to complete after the cheque accepted by the bank. Cancel Transaction Credit / Debits payments made in cheques are set as ‘*pending*’ when added, means still not confirmed by the banks, can be canceled if returned.

Use case Name	Receive Debit
Actors	Manager
Description	Manager adds a debit received from a dealer(debtor).
Pre-conditions	Manager must log in to the system.
Main Flow	
<ol style="list-style-type: none"> <li>1. Manager navigates to ‘Creditors &amp; debtors’ sub section.</li> <li>2. System displays list of creditors and debtors with amount.</li> <li>3. Manager clicks receive debit button in front of particular debtor.</li> <li>4. System displays a form to fill.</li> <li>5. Manager fills and submits.</li> <li>6. System validates, updates the balance and confirms.</li> </ol>	
Alternate Flows	
<p><b>6.1 One or more field is invalid.</b></p> <p>6.1.1 System shows error(s) and prompts to correct.</p>	
Post-conditions	The debit is added successfully.

Table B.4 Use case narrative for Receive Debit

## Web Module

<b>Use case Name</b>	Place Order
<b>Actors</b>	Customer
<b>Description</b>	Customer places order online through the front-side website
<b>Pre-conditions</b>	None
<b>Main Flow</b>	
<ol style="list-style-type: none"> <li>1. Customer selects items to order.</li> <li>2. System adds items to customer's shopping cart.</li> <li>3. Customer views the shopping cart to continue.</li> <li>4. System displays the shopping cart.</li> <li>5. Customer asks the system to place the order.</li> <li>6. System add the order to the customer's orders and confirms.</li> </ol>	
<b>Alternate Flows</b>	
<p><b>5.1 Customer has not logged into system</b>  5.1.1 System asks to login. The use case "customer Login" is executed.</p> <p><b>5.2 Customer is a new customer</b>  5.2.1 System asks to register. The use case "Register Customer" is executed.</p>	
<b>Post-conditions</b>	Customer places an order successfully.

Table B.5 Use case narrative for Place Order

## Reports Module

Stock report is continuing books and discontinued books has some stock are selected for the report. Discontinued books shown in \* with the SKU and low stock, continuing books shown in bold. Sales report can generate the report for a year / annual (month- ALL), month, category and dealer. The canceled categories and dealers are shown in red in the list. Purchase report criteria is similar to sales, agents instead dealers.

<b>Use case Name</b>	Generate Sales Reports
<b>Actors</b>	none
<b>Description</b>	Manager generates the sales reports.
<b>Pre-conditions</b>	The use case "Generate Reports" must be executed.
<b>Main Flow</b>	
<ol style="list-style-type: none"> <li>7. Manager specifies the criteria in sales report section.</li> <li>8. Manager clicks 'Generate Sales Report' button.</li> </ol>	
<b>Alternate Flows</b>	
none	



Post-conditions	The sales report is generated.
-----------------	--------------------------------

Table B.6 Use case narrative for Generate Sales report

Use case Name	Generate Stock Report
Actors	none
Description	Manager generates the stock report.
Pre-conditions	The use case “Generate Reports” must be executed.
Main Flow	
1. Manager clicks ‘Generate Stock Report’ button.	
Alternate Flows	
none	
Post-conditions	The stock report is generated.

Table B.7 Use case narrative for Generate Stock System

## Administration Module

Four user roles in WBBM as follows

- **Super admin**

This is a virtual admin with full access for all modules. However only recommend to create new admin. Then admin create operators and managers. Super admin account cannot be edited, reactivated, deactivated, reset its password, edit profile itself through the application. Direct database editing is needed for those.

- **Admin**

Only allow to access administration module. Admin cannot be modified or create another admin. Admin is only allowed to create and modify lower levels, which are manager and operator

- **Manager**

Allow to access sales, stock, purchase, financial and report modules.

- **Operator**

Allow to access sales and purchase modules

Inactive users are shown in red-colored-text. All most all the activities are recorded in the system log. Cancel, reactivate, edit categories, Agents and dealers not record, because can reactivate and no critical effect. Generating business documents, such as sales invoice not recorded, because it is a part of another process which is recorded.

Use case Name	Backup Data
Actors	Admin
Description	Admin gets current data backup.
Pre-conditions	Admin must logs in to the system
Main Flow	
<ol style="list-style-type: none"> <li>1. Admin navigates to the ‘maintenance’ section of the system.</li> <li>2. System displays the section with backup option.</li> <li>3. Admin clicks ‘Generate backup’ button.</li> <li>4. System generates backup and prompts to save.</li> <li>5. Admin saves the backup in particular location.</li> </ol>	
Alternate Flows	
none	
Post-conditions	The backup is generated successfully.

Table B.8 Use case narrative for Backup Data

Use case Name	Edit Profile
Actors	User
Description	User changes the password.
Pre-conditions	User must logs in to the system
Main Flow	
<ol style="list-style-type: none"> <li>1. User clicks ‘my profile’ button on the main menu.</li> <li>2. System displays user’s profile page.</li> <li>3. User changes the password and confirms through email.</li> <li>4. System updates the password and confirms to email.</li> </ol>	
Alternate Flows	
<p><b>3.1 One or more field is invalid.</b></p> <p>3.1.1. System displays error and prompts to correct.</p>	
Post-conditions	The user’s password is updated successfully through user email.

Table B.9 Use case narrative for edit Profile

**Administration Module**

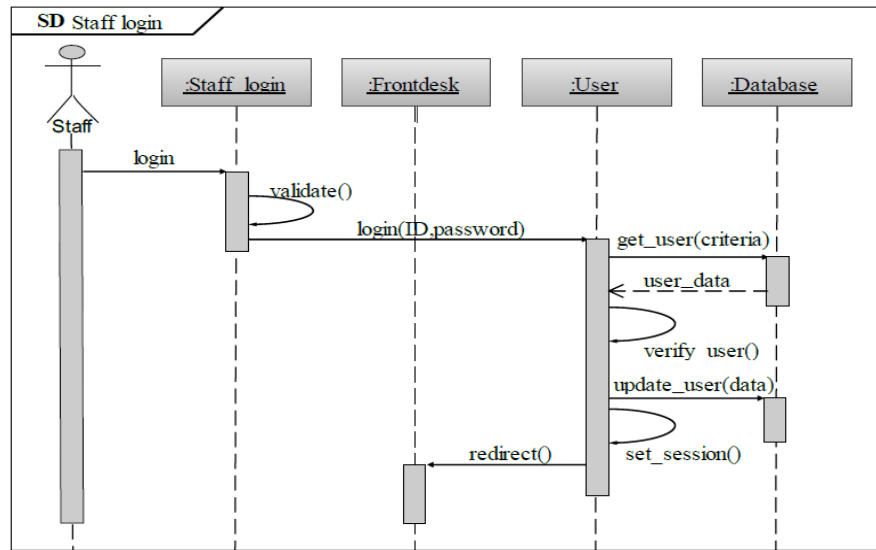


Figure B4 Sequence diagram for Admin Module

The above figure B4 shows Sequence diagram for Admin process.

**Deployment Diagram**

The below Figure B5 Deploy diagram shows the physical architecture of the hardware and software in the system.

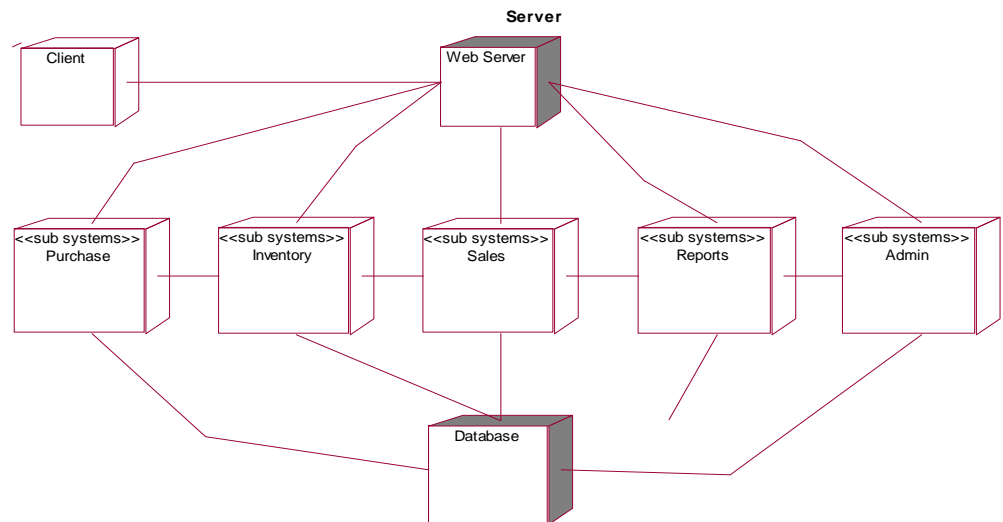


Figure B.5 Deploy diagram of the WBBM

## Structure of the Classes of WBBM

The following figure B.6 represents the properties and methods of the classes of WBBM.

<b>User</b>	<b>Category</b>	<b>Sales Return</b>	<b>Purchase Order</b>
-NIC:Integer -firsrt_name:String -last_name:String -user_role: String -designation: String -last_login: Date Time -status: Boolean  +add_user() +edit_user(id) +search_user(key) +reset_psw (id) +deactivate_user(id) +reactivate_user(id)	-id:Integer -name:String -status:Boolean +add_category() +cancel_category(id) +reactivate_category(id)	-return_id:Integer -sales_id:integer -ISBN:String -quantity_returned:Integer -quantity_provided:Integer -user_added:Integer -date_added:Date Time -user_completed:Integer -date_completed:Date -return_status:String +add_return() +edit_return(id) +complete_return(id) +search_return(key)	-order_id:Integer -agent_id:Integer -date_ordered:Date -user_ordered:Integer -books:Array -order_status:Boolean  +add_order(id) +complete_order(id) +cancel_order(id) +view_order(id) +print_order(id) +search_orders(key)
<b>Log</b>	<b>Agent</b>	<b>Purchase Return Order</b>	<b>Purchase Order</b>
-log_id:Integer -category:String -description:String +add_entry() +search(key)	-agent_id:Integer -name:String -contact:String -status:Boolean +add_agent() +edit_contact(id) +cancel_agent(id) +reactivate_agent(id)	-return_id:Integer -date_ordered:Date Time -user_ordered:Date Time -date_cmp:Date Time -user_cmp: Date Time -sales_returns:Array -order_status:String +add_order() +cancel_order(id) +complete_order(id) +search_order(key) +generate_return_order()	-purchase_id:Integer -invoice_no: String -agent_id:Integer -order_id: Integer -discount:Float -date_purchased:Date -user_purchased:Integer -books:Array +add_purchase() +view_purchase(id) +print_grn(id) +search_purchase(key)
<b>Book</b>	<b>Dealer</b>	<b>Requisition</b>	<b>Backup</b>
-ISBN:String -title:String -publisher:String -agent_id:Integer -category_id:Integer -list_price:Float -purchase_price:Float -discount:Float -min_stock:Integer -max_stock:Integer -current_stock:Integer -is_ordered:Boolean -status:Boolean +add_book() +edit_book(id) +discontinue_book(id) +search_book(key)	-dealer_id:Integer -name:String -contact:String -status:Boolean +add_agent() +edit_contact(id) +cancel_dealer(id) +reactivate_dealer(id)	-request_id:Integer -note:String -date_added:Date Time -user_added: Integer -date_com_can:Date -user_com_can:Integer -status:String +add_request() +cancel_request(id) +complete_request(id)	+get_backup()
	<b>Sales</b>		<b>Report</b>
	-sales_id:Integer -dealer_id:Integer -date_sold: Date Time -user_sold: Integer -books_sold:Array  +add_sales() +view_sales(id) +search_sales(key) +generate_invoice(id)		+generate_report(data)
	<b>Stock</b>		
	-books:array		

Figure B.6 Classes of the WBBM

## Architectural Design

Architectural Design is a way to establish the overall structure of a Software System. It is the design process for identifying the sub-systems making up the framework for sub-system and communication.

The following figure B.7 illustrates the components of WBBM using a Component Diagram.

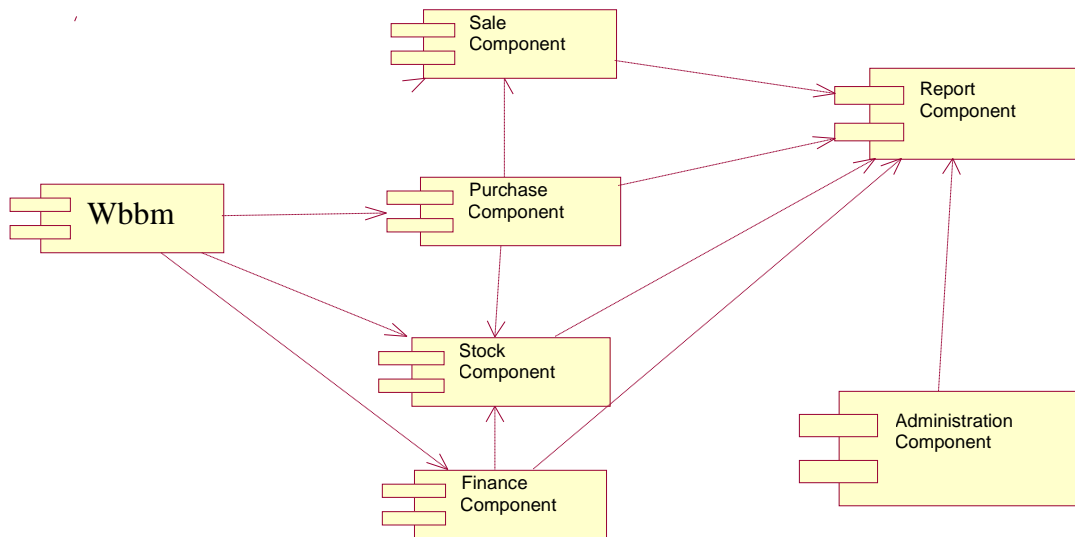


Figure B.7 Components of the WBBM

## Detail Structure of the WBBM Database Design

More details of the WBBM database design are given in the following figures.

The following figure B.8 represents the database structure related to the purchase and sale modules.



Figure B.8 Database structure for the WBBM

## Appendix C | User Documentation

Users who have privileges to access the system about how to navigate within the system and how to use its functions are helped by this document. Eligibility to access the system is available to manager, accountant, administrator, wholesale dealer and sales representative.

Different user levels are available to each and every member and different profile pages are provided for their management. The administrator's profile which has all the system privileges is described by this document. This documentation provides you a guidance to use the backend web application efficiently.

### Login page and other functionalities



Open the web browser that configured in the installation.

Type the URL in the address bar of the browser ***http://localhost/wbbm/*** and navigate. After navigation, user can find the Login page that is under figure C.1.

Figure C.1 Application Login

After login, by entering username (NIC) and password it directs user to the page which is under figure C.2

The following figure C.2 shows relevant personnel information and additional information can be identified by the user. To view more additional information, user should activate More Information link. Login Time section provides the login information.

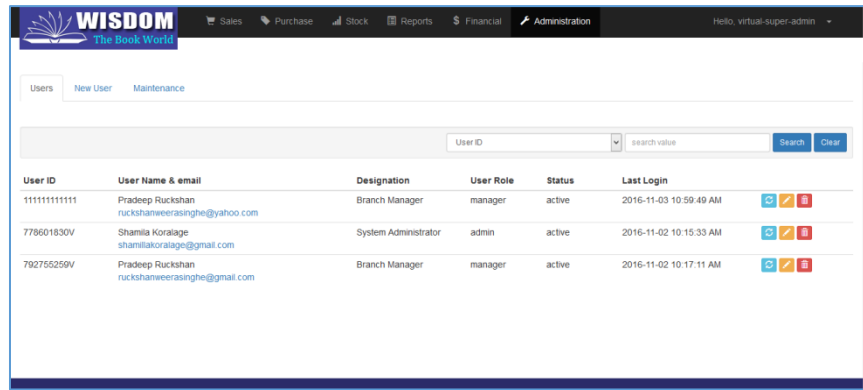


Figure C.2 Administrator Home page

### Change Password Facility

By clicking, link which is located in the figure C.2 the current password can be changed by the user. After clicking that link the pop up appears which is shown under figure C.3 in password reset.

Current user’s password can be changed by entering new password accurately into the fields.

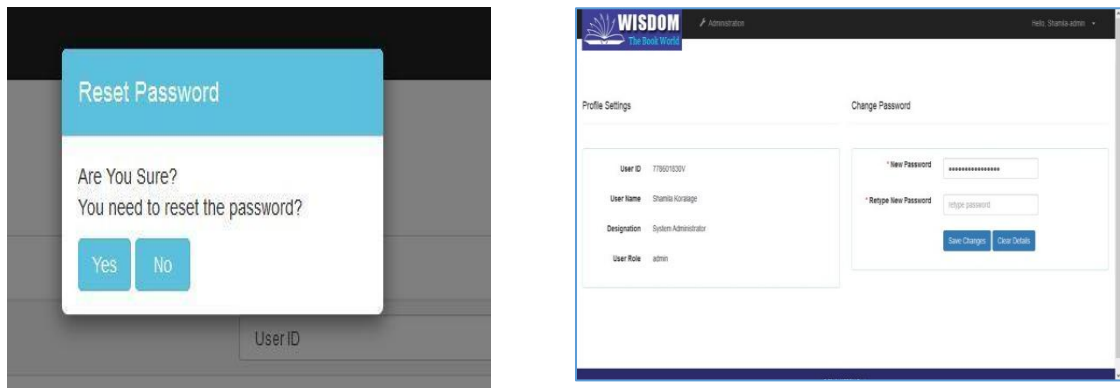


Figure C.3 Password Reset page

Administration module creates username and passwords to users then after login to system they can reset their password for only special privilege user roles such as manager and accountants.

### Navigation between Pages

To navigate in between system pages, link bar which is mentioned below figure C.4 can be used. (Please refer figure C.2 to view actual position of figure C.4).



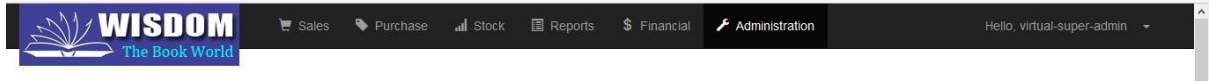


Figure C.4 Link Bar

### Logout from the system

To logout from the system, the link can be found on the top right hand corner of the figure C.5. After clicking this link user can successfully logout from the system and it will direct user back to the Login Page.

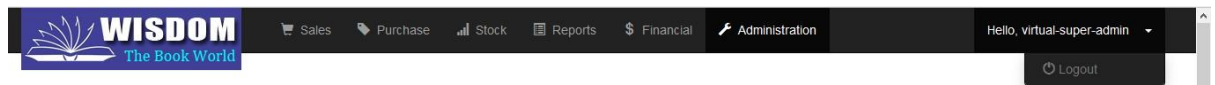


Figure C.5 Logout Area.

### Data Manipulation in Tables and Operations

This is a common operation which comes with more pages. Please see the figure C.6 to get an idea about it and the arrow heads under the figure.

This belongs to link. (Note:- This facility is met in many pages of the system and this introduction will help to get an idea.) The figure C6 shows Book data details.

SKU	Category	Title, ISBN, Author & Publisher	Agent & Purchase Price	Price & Discount	Min - Max Current Stock	Last Sold	Status	Ordered	Added
11	general-english	Advanced English Grammer 1111111111115 Raymond Murphy Cmabridge University Press	Vijitha Yapa Bookshop 350.00	500.00 0%	10 - 50 0	-	discontinued	Yes	- 2016-10-20
13	edexcel-maths-oi	Anotehr Test 9999999999990 Raymond Murphy Sdsdsdsds	Sarasavi Bookshop 200.00	250.00 0%	10 - 45 0	-	continue	No	2016-10-19 -
12	edexcel-maths-oi	Book For Chart 9999999999999 Sdsdsdsdsd Sdsdsdsds	Sarasavi Bookshop 100.00	100.00 0%	50 - 100 15	2016-10-19	continue	No	2016-02-10 -
8	edexcel-science-oi	Edexcel Science Class Text	Vijitha Yapa Bookshop	3500.00	50 - 100	-	discontinued	No	2016-09-01

Figure C.6 Table view of data.

## Working with Data Grid



Figure C.7 Data Grid

As you see in figure C.7 Data grid consists of several areas each of which is explained below.

- **A – Clear Search button.** Click this button to cancel the current search you made and reload the default data set.
- **B – Search Area.** You can use this area to filter the data under the various criteria.
- **C– Data Area.** This area shows the current data set in tabulating format. Note that it usually provides the most needed data only. You have to *view* the particular record to if you need to see complete data of that record.
- **D – Action Buttons.** This buttons allows functions to manipulate the particular record.

## Working with Searching Area

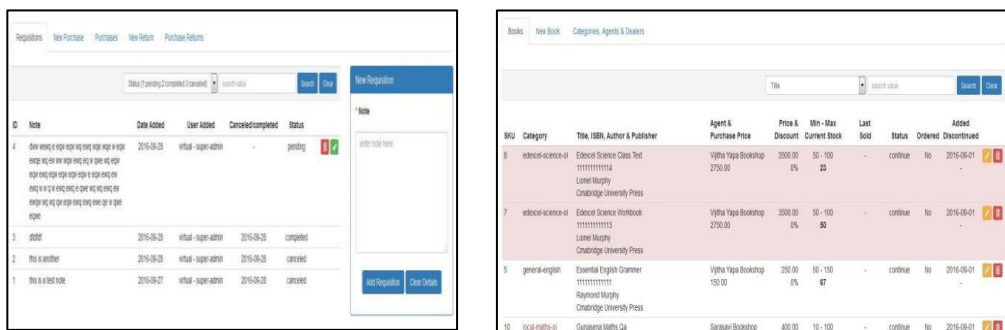


Figure C.8 Search Area

As shown above figure C.8, search area allows you to filter the data under a certain criterion.

To filter the data,

- Select the search key from the dropdown list. The options provided depend on the particular section.
- Type the search value in to the search box.
- Press *search* button.

For some options, the value should be one from the predefined value set. In such cases hint about the value is provided within the option.

## Working with Action Buttons

Action buttons provides specific functions you can do on the particular record, such as view details and edit records. Most of the buttons are common for the whole application, while some specific buttons available for specific function related to that particular area.

Following figure C.9 represents the activity buttons used for common activities.

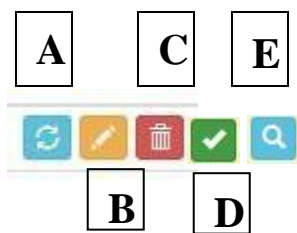


Figure C.9 Action Buttons

- **A – Reset Button.** Click this button to change details of the relevant record.
- **B – Edit Button.** Click this button to display data in edit mode to make changes.
- **C – Cancel Button.** Click this button to cancel/remove particular record.
- **D – Process Button.** This represents particular process depends on the section.  
Ex: process sales order, process purchase order, etc.
- **E – View Button.** Click this button to view full details of the relevant record.

## Working with Forms

Forms are used throughout the application to add new data and to edit existing data. The following figure C.10 shows some form elements.

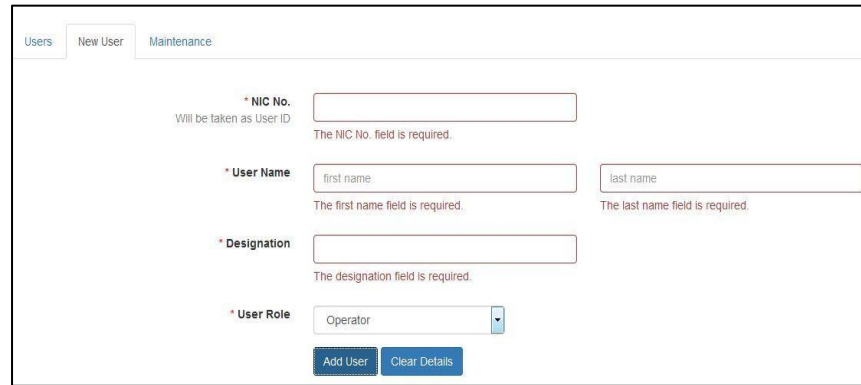
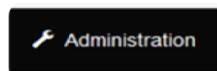


Figure C.10 Form Notifications

## Administration Page



A major topic which comes with the system is the Administration Page. This page only appears to administrator. After navigating to this page, a data manipulation table can be seen by the user which is described under figure C.11. There is a button Add New Users in the page. After clicking that button it can find following forms and user can manage user details by using these forms.

### • Personnel Information Page

According to user levels new user's information can be added to the system by the manager. When selecting a job title, the system automatically enables and disables relevant fields.

The following figure C11 shows enter personal data to system of new user.

(Path: Administration → New user → Personnel Information)

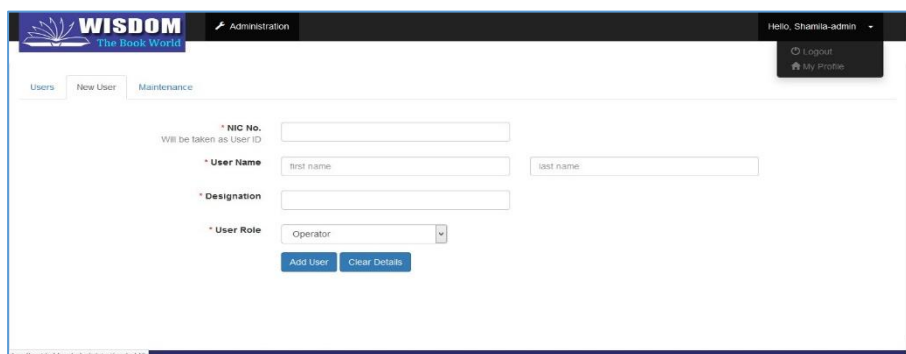
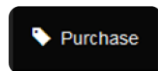


Figure C.11 Adding Personnel Information for new user

Use common form elements such as text boxes and dropdown lists to enter the data.

- Press **Clear** button to reset the form by clearing all the changes you made.
- Select user role drop down menu from drop down list users.
- Press **Add user button** to confirm and process data. The button can be changed as **OK, Add, Save** and so on according to the purpose of the form.

## Purchase Page



This is another major system page that can be seen inside the link bar. When handling this page administrator is the responsible person.

- **Add New Purchase**

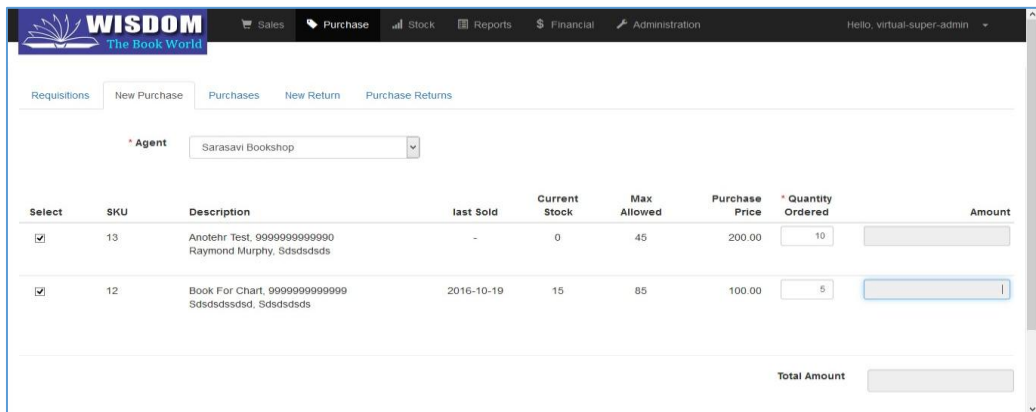


Figure C.12 Add New Purchase

- **View Purchase**

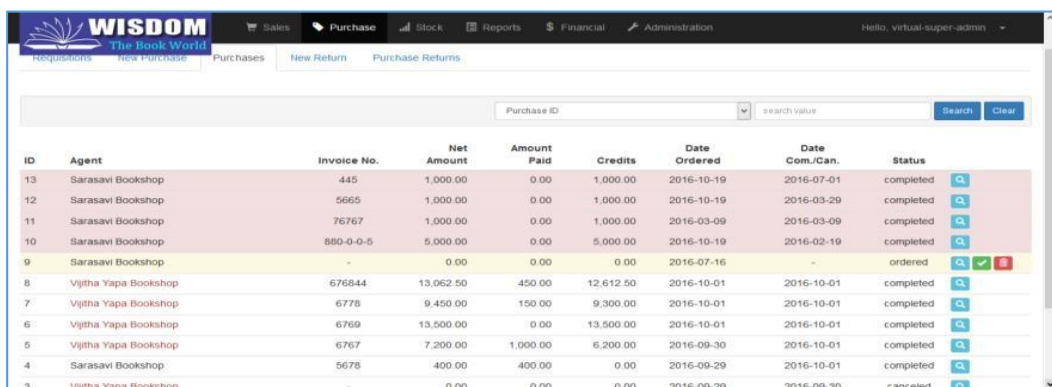
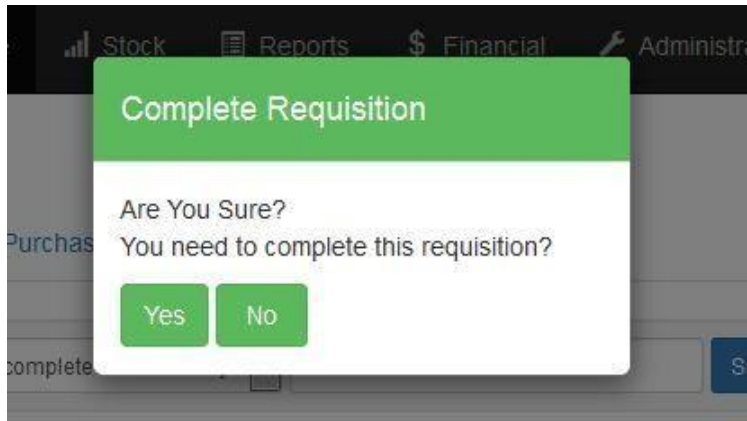


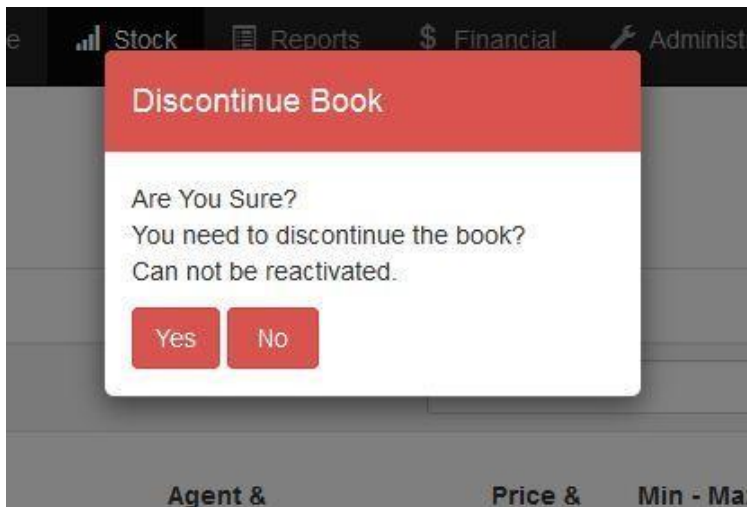
Figure C.13 View of Purchase Page in WBBM

- **Processing Purchasing Requests**

This shows you to process purchasing requests into a purchasing order.



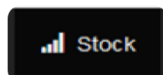
- \*Select **Purchasing Management**
- \***Purchasing Requests**
- \* **Process Purchasing Requests**



- \* Select one or more request to process.
- \* Select the Stock.
- \* Press **process**.
- \* You can use **Search Book** area to search books if needed.

Figure C.14 Process of Purchase requisition and discontinue Notifications in WBBM

## Stock Page



Stocks are mainly maintained under the stock page. Similar operations such as add stock and view stock are common to both. The following are displayed in biscuit stock handling page by considering those similarities.

- **Add Stock**

Adding new book stocks, updating book stocks and removing stocks are done using this form. Other update if the user wants to add a new stock.

The figure C15 shows new book details enter to system. (Path: Stock → New Book → Add Book)

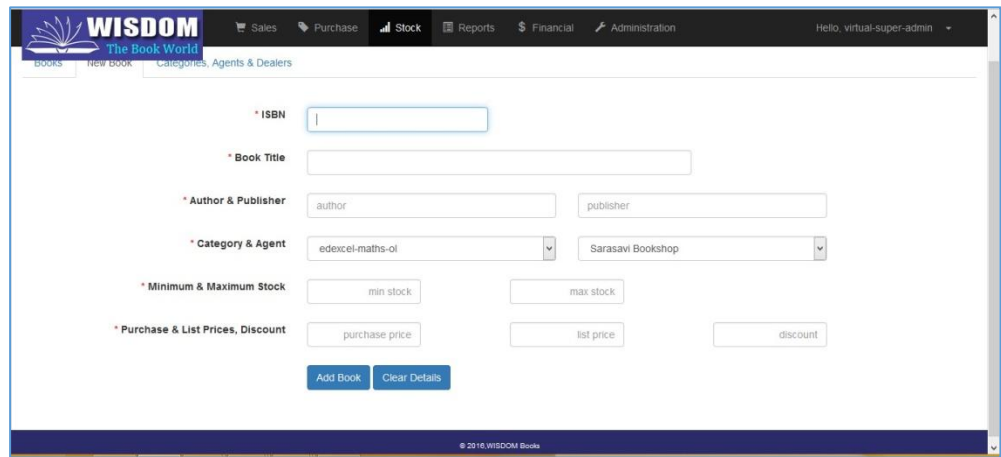


Figure C.15 Add Books to Stock in WBBM

- **View Stock**

The current book stock can be easily identified by the user using this page (refer figure C.16). New book stock if added to this table with a View Details Link. (Note:- Dummy data).

The details of the selected book stock shows in figure C16 when the user click on the link. (Path: Stock → Book Stock → View Stock)

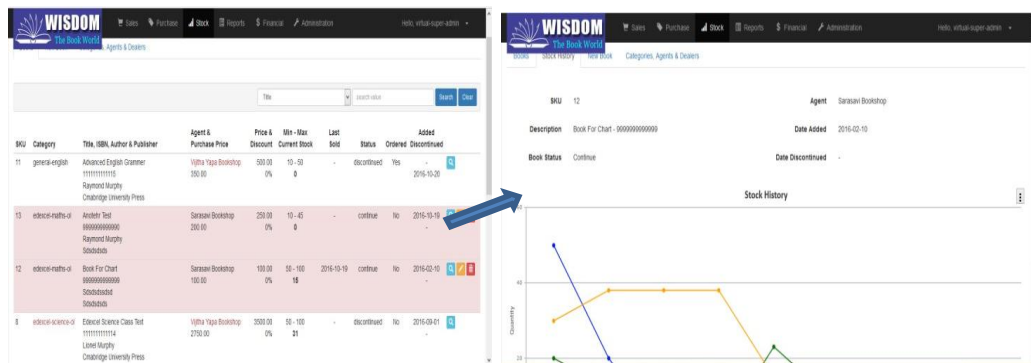


Figure C.16 View of Books Stock in WBBM

## Appendix D | Management Reports

The WBBM generates a comprehensive set of documents during its execution which is one of the objectives of the WBBM as well. Some of those provide summarized business data needed to the management decision making process while others are the business related documents such as purchase and sales invoices.

The following section represents the reports generated by the WBBM for the purpose of Management of the business.

### Monthly Report

- Sales

As shown in figure D.1, this report consists of the amount of Sales per month.

WISDOM BOOK SHOP		Monthly Sales Report		
Negombo Road, Wattala, Sri Lanka 011 2948586, wisdom@sitnet.lk, www.wisdombooks.lk				
Duration:	2017 - November			Date: 2017-11-02 21:36:19
Category:	ALL			User: virtual-super-admin
Dealer:	ALL			
SKU	Title	Quantity	Amount Listed	Amount Sold
1	Click Start Computer Science For Schools - 1111555596789	5	2,725.00	2,725.00
4	Cambridge Biology O/L - 97804350123	3	11,040.00	10,672.00
5	Cambridge Eoconomics O/L - 9780435012	2	10,320.00	9,288.00
10	Cambridge Core Mathematics For Cam IGCSE - 4563458974	2	4,164.00	4,164.00
12	Cambridge Active Spelling - 6 - 2345168913	2	762.00	762.00
<b>Total</b>		<b>14</b>	<b>29,011.00</b>	<b>27,611.00</b>

Figure D1 Monthly Sales Report details

- Stock

As shown in figure D.2, this report provides the details of the current stock per month for one Agent. The products which have a stock below the threshold level have been highlighted as well.

WISDOM BOOK SHOP		Stock Report						
Negombo Road, Wattala, Sri Lanka 011 2948586, wisdom@sitnet.lk, www.wisdombooks.lk								
Category:	Click Start Computing - Grade 1						Date: 2017-10-02 12:25:12	
Agent:	Expographic Books Pvt Ltd						User: R-manager	
SKU	Title	Medium	Min-Stock	Max-Stock	Last Sold	Cur-Stock	List Price	Amount
1	Click Start Computer Science For Schools		10	25	2017-12-02	3	545.00	1,635.00
<b>Total</b>						<b>3</b>	<b>-</b>	<b>1,635.00</b>

Figure D2 Monthly Stock Report details per one Agent



## Annual Reports

- Sales

As shown in figure D.3, this report provides the details of all the sales made in the entire year, categorized according to the product.

**WISDOM BOOK SHOP**  
 Negombo Road, Wattala, Sri Lanka  
 011 2948586, wisdom@sitnet.lk, www.wisdombooks.lk

**Annual Sales Report**


Duration: 2017 - ALL  
 Category: ALL  
 Dealer: ALL  
 Date: 2017-11-02 21:37:46  
 User: virtual-super-admin

Month	SKU	Title	Quantity	Amount Listed	Amount Sold
<b>January</b>	No sales				
<b>Sub-Total</b>			0	0.00	0.00
<b>February</b>	No sales				
<b>Sub-Total</b>			0	0.00	0.00
<b>March</b>	No sales				
<b>Sub-Total</b>			0	0.00	0.00
<b>April</b>	No sales				
<b>Sub-Total</b>			0	0.00	0.00
<b>May</b>	No sales				
<b>Sub-Total</b>			0	0.00	0.00
<b>June</b>	No sales				
<b>Sub-Total</b>			0	0.00	0.00
<b>July</b>	No sales				
<b>Sub-Total</b>			0	0.00	0.00
<b>August</b>	4	Cambridge Biology O/L - 97804350123	2	7,360.00	6,992.00
	10	Cambridge Core Mathematics For Cam IGCSE - 4563458974	5	10,410.00	9,369.00
<b>Sub-Total</b>			7	17,770.00	16,361.00
<b>September</b>	2	Click Start Computer Science For Schools - 122333300	5	2,725.00	2,643.25
	3	Cambridge Biology O/L - 97804350123	1	3,680.00	3,680.00
	4	Cambridge Biology O/L - 97804350123	5	18,400.00	18,400.00
<b>Sub-Total</b>			11	24,805.00	24,723.25
<b>October</b>	No sales				
<b>Sub-Total</b>			0	0.00	0.00
<b>November</b>	1	Click Start Computer Science For Schools - 1111555596789	5	2,725.00	2,725.00
	4	Cambridge Biology O/L - 97804350123	3	11,040.00	10,672.00
	5	Cambridge Economics O/L - 9780435012	2	10,320.00	9,288.00
<b>Sub-Total</b>			10	24,085.00	22,685.00
<b>Total</b>			49	94,371.00	90,911.15

Figure D3 Annual Sales Report details of WBBM

- Purchase

As shown in figure D.4, this report provides the details of all the purchase made in the entire year, categorized according to the product.



**WISDOM BOOK SHOP**  
 Negombo Road, Wattala, Sri Lanka  
 011 2948586, wisdom@sitnet.lk, www.wisdombooks.lk

---

**Annual Purchase Report**

Duration: 2017 - ALL  
 Category: ALL  
 Agent: ALL  
 Date: 2017-11-02 21:41:17  
 User: virtual-super-admin

SKU	Title	Quantity	List Amount	Purchased Amount
<b>January</b>				
No Purchases				
<b>Sub-Total</b>		0	0.00	0.00
<b>February</b>				
No Purchases				
<b>Sub-Total</b>		0	0.00	0.00
<b>March</b>				
No Purchases				
<b>Sub-Total</b>		0	0.00	0.00
<b>April</b>				
No Purchases				
<b>Sub-Total</b>		0	0.00	0.00
<b>May</b>				
No Purchases				
<b>Sub-Total</b>		0	0.00	0.00
<b>June</b>				
No Purchases				
<b>Sub-Total</b>		0	0.00	0.00
<b>July</b>				
No Purchases				
<b>Sub-Total</b>		0	0.00	0.00
<b>August</b>				
3	Cambridge Biology O/L - 97804350123	2	5,500.00	5,225.00
<b>Sub-Total</b>		2	5,500.00	5,225.00
<b>September</b>				
1	Click Start Computer Science For Schools - 1111555596789	15	6,750.00	6,750.00
2	Click Start Computer Science For Schools - 122333300	10	4,500.00	4,500.00
<b>Sub-Total</b>		25	11,250.00	11,250.00
<b>October</b>				
4	Cambridge Biology O/L - 97804350123	10	27,500.00	27,500.00
5	Cambridge Economics O/L - 9780435012	12	55,800.00	55,800.00
8	Edexcel GCE In Applied ICT A2 - 1325689637894	5	28,875.00	28,875.00
10	Cambridge Core Mathematics For Cam IGCSE - 4563458974	8	14,048.00	14,048.00
12	Cambridge Active Spelling - 6 - 2345168913	5	1,225.00	1,163.75
14	Cambridge Commerce - 2013400678	10	17,560.00	16,682.00
<b>Sub-Total</b>		50	145,008.00	144,068.75
<b>November</b>				
11	Cambridge Advanced Learning Dictionary 3rd Edition - 4125745623	10	16,250.00	15,437.50
15	Cambridge Accounting O/L - 4561237895	6	10,536.00	10,536.00
<b>Sub-Total</b>		16	26,786.00	25,973.50

Figure D.4 Annual Purchase Report details of WBBM

• Stock

stock\_report (2).pdf 1 / 2

**WISDOM BOOK SHOP**  
 Negombo Road, Wattala, Srilanka  
 011 2948586, wisdom@sitnet.lk, www.wisdombooks.lk

**Stock Report**  
 Category: ALL Date: 2017-11-02 11:07:47  
 Agent: ALL User: R-manager

SKU	Title	Min-Stock	Max-Stock	Last Sold	Cur-Stock	List Price	Amount
1	Click Start Computer Science For Schools	10	25	2017-12-02	3	545.00	1,635.00
2	Click Start Computer Science For Schools	10	25	2017-11-09	5	545.00	2,725.00
3	Cambridge Biology O/L	5	15	2017-11-09	1	3,680.00	3,680.00
4	Cambridge Biology O/L	10	25	2017-10-10	0	3,680.00	0.00
5	Cambridge Economics O/L	10	25	2017-11-18	10	5,160.00	51,600.00
7	Edexcel IGCSE Chemistry	10	25	-	0	5,160.00	0.00
8	Edexcel GCE In Applied ICT A2	5	25	-	5	6,375.00	31,875.00
9	Edexcel Biology For AS	12	25	-	0	3,150.00	0.00
10	Cambridge Core Mathematics For Cam IGCSE	10	25	2017-12-09	1	2,082.00	2,082.00
11	Cambridge Advanced Learning Dictionary 3rd Edition	5	15	2017-12-02	0	1,897.00	0.00
12	Cambridge Active Spelling - 6	10	20	2017-11-09	10	381.00	3,810.00
13	Cambridge Advanced Chemistry	10	20	-	0	2,625.00	0.00
14	Cambridge Commerce	10	30	-	10	2,275.00	22,750.00
15	Cambridge Accounting O/L	5	15	-	6	2,275.00	13,650.00
16	Cambridge Active Spelling -	5	10	-	0	900.00	0.00
17	Cambridge Active Spelling	5	10	-	2	380.00	760.00
19	Cambridge Advanced Chemistry Cambridge	5	15	-	0	2,625.00	0.00
20	Cambridge Advanced Chemistry	5	15	-	0	2,625.00	0.00
21	Cambridge Biology O/l	5	8	-	0	4,250.00	0.00
22	Kumararodaya	5	20	-	0	525.00	0.00
23	Asss	5	15	-	0	450.00	0.00
<b>Total</b>					<b>53</b>	<b>-</b>	<b>134,567.00</b>


Activate Windows  
Go to PC settings to activate Windows.

Figure D.5 Stock Category of WBBM

## Purchase Order

Purchase orders are generated when making the purchasing from the particular supplier.

Structure of the Purchase Order is given in figure D.6



**WISDOM BOOK SHOP**  
Negombo Road, Wattala, Sri Lanka  
011 2948586, wisdom@sitnet.lk, www.wisdombooks.lk


---

**Purchase Order**

Order ID: **000020**      Ordered by: R Pradeep, Branch Manager  
 Ordered from: Expographic Books Pvt Ltd      Ordered on: 2017-08-02

SKU	Title	List Price	Quantity	Amount
17	Cambridge Active Spelling , 62345168913 Cambridge, Cambridge	278.00	2	556.00
<b>Total</b>			<b>2</b>	<b>556.00</b>

Figure D.6 Purchase Order of WBBM



**WISDOM BOOK SHOP**  
Negombo Road, Wattala, Sri Lanka  
011 2948586, wisdom@sitnet.lk, www.wisdombooks.lk

---


**GRN [Order ID: 000019]**

Invoice No.: 56465623      Ordered on: 2017-12-01  
 Ordered from: Jeya Book Centre - Pettah      Received on: 2017-11-03

SKU	Title	Price Ordered	Quantity Ordered	Amount Ordered	Price Purchased	Quantity Purchased	Amount Purchased
20	Cambridge Advanced Chemistry , 4561237896 Peter Robinson, Cambridge	1,950.00	5	9,750.00	1,950.00	5	9,750.00
<b>Total</b>			<b>5</b>	<b>9,750.00</b>		<b>5</b>	<b>9,750.00</b>
<b>Net Amount (Discount: 0%)</b>							<b>9,750.00</b>

Activate Windows

Figure D.7 Purchase GRN of WBBM



**WISDOM BOOK SHOP**  
Negombo Road, Wattala, Sri Lanka  
011 2948586, wisdom@sitnet.lk, www.wisdombooks.lk

---

**Purchase Return Order**

Order ID: **000004**      Ordered by: R Pradeep, Branch Manager  
 Ordered from: Expographic Books Pvt Ltd      Ordered on: 2017-10-08

SKU	Title	Quantity
1	Click Start Computer Science For Schools - 1111555596789	1
<b>Total</b>		<b>1</b>

Figure D.8 Purchase Order Return of WBBM

## Sales Order

Sales orders are generated when making the Sales from the particular customer.

Structure of the Sales order is given in figure D.9

WISDOM BOOK SHOP					
Negombo Road, Wattala, Srilanka 011 2948586, wisdom@slt.net.lk, www.wisdombooks.lk					
Sales Invoice					
Invoice No:	<b>000018</b>	Cashier: virtual			
		2017-11-03			
SKU	Title	Price	Quantity	Discount	Amount
1	Click Start Computer Science For Schools - 1111555596789	545.00	3	0%	1,635.00
<b>Total</b>			<b>3</b>		<b>1,635.00</b>
Discount				0%	0.00
<b>Net Amount</b>					<b>1,635.00</b>
Cash					2,000.00
<b>Balance</b>					<b>365.00</b>

Figure D.9 Sales Order of WBBM

WISDOM The Book World							
Sales							
ID	Type	Dealer	Net Amount	Amount Paid	Debits	Date Sold	Status
10	whole	Gateway International	15,000.00	15,000.00	0.00	2017-08-05	completed
9	retail	-	2,700.00	2,700.00	0.00	2017-08-05	completed
8	retail	-	28,975.00	28,975.00	0.00	2017-07-20	completed
7	whole	Lyceum International	12,375.00	3,000.00	9,375.00	2017-07-19	completed
6	retail	-	7,425.00	7,425.00	0.00	2017-07-19	completed
5	online	-	31,875.00	875.00	31,000.00	2017-05-29	completed
4	retail	-	2,505.00	2,505.00	0.00	2017-06-29	completed
3	whole	Gateway International	41,250.00	41,250.00	0.00	2017-06-24	completed

Figure D.10 Sales details of WBBM

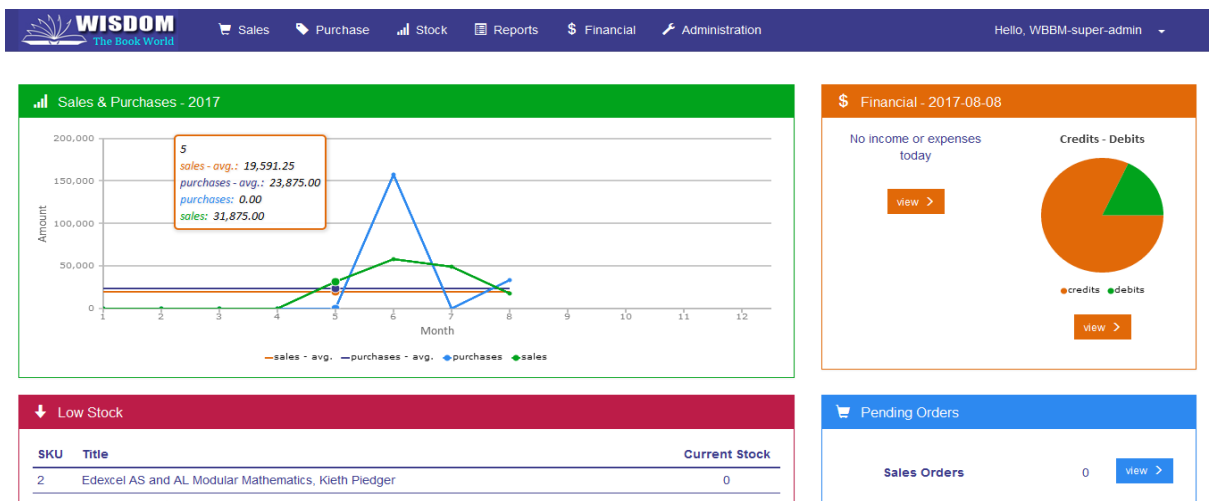


Figure D.11 Sales, Purchase information of a month in WBBM

## Report Types generate from WBBM

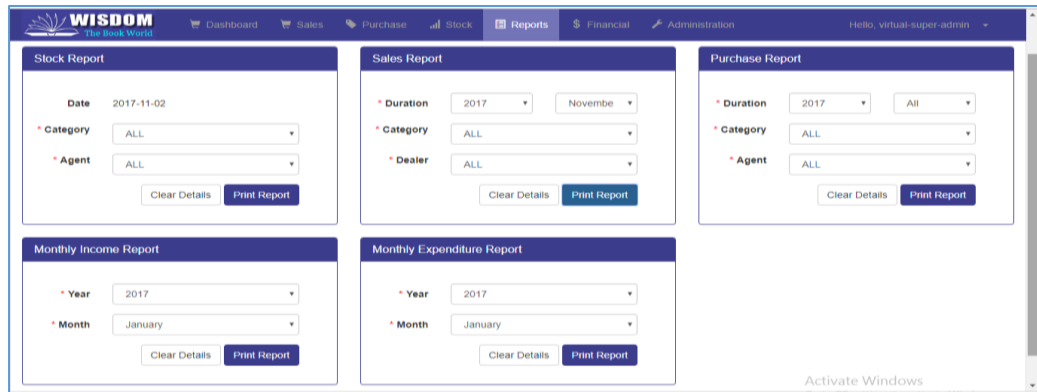


Figure D12. Stock, Sales, Purchase, Income, Expenditure Reports

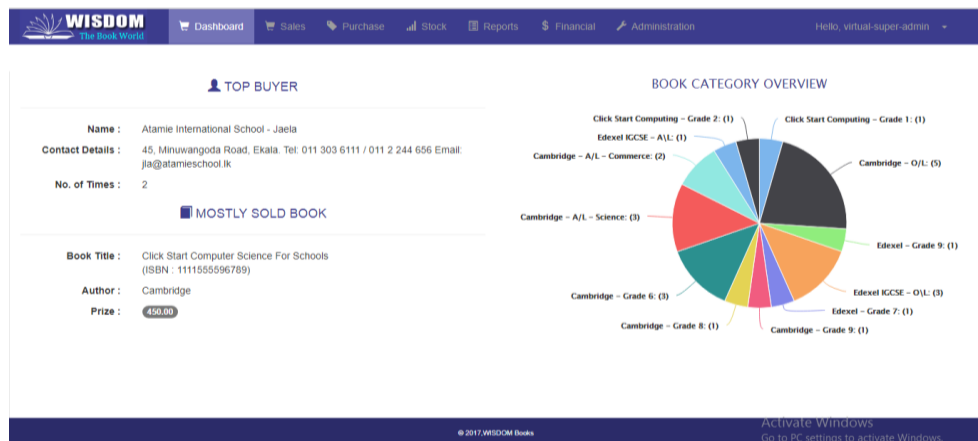


Figure D13. Summary of Stock for a day

## Payment Invoice Report

WISDOM BOOKSHOP		Payment Invoice		
Negombo, Sri Lanka 011 294858, wisdom@stinet.lk, www.wisdombooks.lk		Order No: 00009	Prepared By: Shamila	
		Agent: All	Prepared On: 2016-07-24	
Date	Description	Payment Type	Payment Method	Amount
2016-07-23	Sales: 6	income	cash	3,000.00
2016-07-23	Pay credit: Sarasavi Bookshop.	expenditure	cash	6,000.00
2016-07-23	receive debit: lyceum.	income	cash	9,500.00
2016-07-22	Sales: 3	income	cash	5,500.00
2016-07-22	Purchase: 2	expenditure	cash	46,700.00
2016-07-22	Sales: 1	income	cash	9,000.00

Figure D.14 Payment Invoice

## Appendix E | Test Results

Following tables from E.1 to E.5 provide the test results for some selected test cases given in the evaluation chapter.

### Purchase Module

Test No.	Actual Result [screen shots]	Status																
1	 <table border="1"> <thead> <tr> <th>Description</th> <th>Date Added</th> <th>User Added</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td>Edexcel GCE AL ICT 6 books for lycecum.</td> <td>2016-08-02</td> <td>778601830V</td> <td>Pending</td> </tr> </tbody> </table>	Description	Date Added	User Added	Status	Edexcel GCE AL ICT 6 books for lycecum.	2016-08-02	778601830V	Pending	ok								
Description	Date Added	User Added	Status															
Edexcel GCE AL ICT 6 books for lycecum.	2016-08-02	778601830V	Pending															
2	<table border="1"> <thead> <tr> <th>Description</th> <th>Date Added</th> <th>User Added</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td>Edexcel GCE AL ICT 6 books for lycecum.</td> <td>2016-08-02</td> <td>778601830V</td> <td>Pending</td> </tr> <tr> <td>Request atlashall Ol history book. 0112443499- hiran</td> <td>2016-07-23</td> <td>778601830V</td> <td>Canceled</td> </tr> <tr> <td>Advanced Eng Grammar-murphy, ordered online- order 2343</td> <td>2016-07-23</td> <td>778601830V</td> <td>Completed</td> </tr> </tbody> </table>	Description	Date Added	User Added	Status	Edexcel GCE AL ICT 6 books for lycecum.	2016-08-02	778601830V	Pending	Request atlashall Ol history book. 0112443499- hiran	2016-07-23	778601830V	Canceled	Advanced Eng Grammar-murphy, ordered online- order 2343	2016-07-23	778601830V	Completed	ok
Description	Date Added	User Added	Status															
Edexcel GCE AL ICT 6 books for lycecum.	2016-08-02	778601830V	Pending															
Request atlashall Ol history book. 0112443499- hiran	2016-07-23	778601830V	Canceled															
Advanced Eng Grammar-murphy, ordered online- order 2343	2016-07-23	778601830V	Completed															
4	<table border="1"> <thead> <tr> <th>Status</th> </tr> </thead> <tbody> <tr> <td>Pending   </td> </tr> <tr> <td>Canceled</td> </tr> <tr> <td>Completed</td> </tr> </tbody> </table>  <table border="1"> <thead> <tr> <th>Status</th> </tr> </thead> <tbody> <tr> <td>Canceled</td> </tr> <tr> <td>Canceled</td> </tr> <tr> <td>Completed</td> </tr> </tbody> </table>	Status	Pending   	Canceled	Completed	Status	Canceled	Canceled	Completed	ok								
Status																		
Pending   																		
Canceled																		
Completed																		
Status																		
Canceled																		
Canceled																		
Completed																		


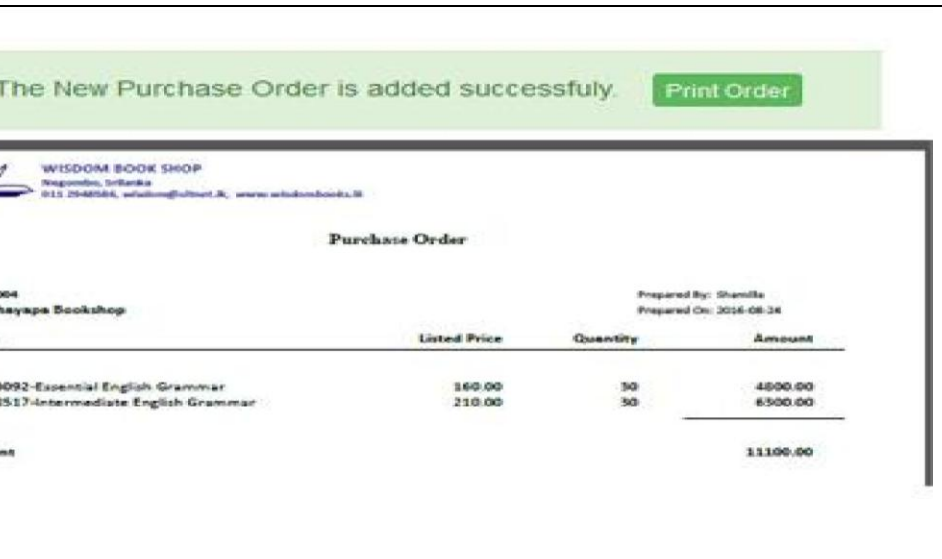


5		ok																												
7	 <table border="1" data-bbox="319 772 1380 1120"> <thead> <tr> <th colspan="4">Purchase Order</th> </tr> <tr> <td>Order No: 00004</td> <td colspan="2">Prepared By: Shanika</td> <td></td> </tr> <tr> <td>Agent: Vijithayapa Bookshop</td> <td colspan="2">Prepared On: 2016-08-24</td> <td></td> </tr> <tr> <th>Description</th> <th>Listed Price</th> <th>Quantity</th> <th>Amount</th> </tr> </thead> <tbody> <tr> <td>9788175963092-Essential English Grammar</td> <td>160.00</td> <td>30</td> <td>4800.00</td> </tr> <tr> <td>9788185618517-Intermediate English Grammar</td> <td>210.00</td> <td>30</td> <td>6300.00</td> </tr> <tr> <td><b>Total Amount</b></td> <td></td> <td></td> <td><b>11100.00</b></td> </tr> </tbody> </table>	Purchase Order				Order No: 00004	Prepared By: Shanika			Agent: Vijithayapa Bookshop	Prepared On: 2016-08-24			Description	Listed Price	Quantity	Amount	9788175963092-Essential English Grammar	160.00	30	4800.00	9788185618517-Intermediate English Grammar	210.00	30	6300.00	<b>Total Amount</b>			<b>11100.00</b>	ok
Purchase Order																														
Order No: 00004	Prepared By: Shanika																													
Agent: Vijithayapa Bookshop	Prepared On: 2016-08-24																													
Description	Listed Price	Quantity	Amount																											
9788175963092-Essential English Grammar	160.00	30	4800.00																											
9788185618517-Intermediate English Grammar	210.00	30	6300.00																											
<b>Total Amount</b>			<b>11100.00</b>																											

Table E.1 Test results for the Purchase Module

### Sales Module

Test No.	Actual Result [screen shots]	Status																														
1	 <table border="1" data-bbox="288 1612 1337 1771"> <thead> <tr> <th>Invoice No</th> <th>Type</th> <th>Dealer</th> <th>Date Sold</th> <th>Amount</th> <th>U:</th> </tr> </thead> <tbody> <tr> <td>11</td> <td>Retail</td> <td>-</td> <td>2016-08-14</td> <td>760.00</td> <td>7;</td> </tr> <tr> <td>8</td> <td>Retail</td> <td>-</td> <td>2016-08-14</td> <td>5045.00</td> <td>7;</td> </tr> <tr> <td>4</td> <td>wholesale</td> <td>Lycium International</td> <td>2016-08-14</td> <td>14500.00</td> <td>7;</td> </tr> <tr> <td>1</td> <td>Retail</td> <td>-</td> <td>2016-08-14</td> <td>150.00</td> <td>7;</td> </tr> </tbody> </table>	Invoice No	Type	Dealer	Date Sold	Amount	U:	11	Retail	-	2016-08-14	760.00	7;	8	Retail	-	2016-08-14	5045.00	7;	4	wholesale	Lycium International	2016-08-14	14500.00	7;	1	Retail	-	2016-08-14	150.00	7;	ok
Invoice No	Type	Dealer	Date Sold	Amount	U:																											
11	Retail	-	2016-08-14	760.00	7;																											
8	Retail	-	2016-08-14	5045.00	7;																											
4	wholesale	Lycium International	2016-08-14	14500.00	7;																											
1	Retail	-	2016-08-14	150.00	7;																											
3	 <table border="1" data-bbox="319 1870 1337 1937"> <thead> <tr> <th>Invoice No</th> <th>Type</th> <th>Dealer</th> <th>Date Sold</th> <th>Amount</th> </tr> </thead> <tbody> <tr> <td>11</td> <td>Retail</td> <td>-</td> <td>2016-08-14</td> <td>760.00</td> </tr> </tbody> </table>	Invoice No	Type	Dealer	Date Sold	Amount	11	Retail	-	2016-08-14	760.00	ok																				
Invoice No	Type	Dealer	Date Sold	Amount																												
11	Retail	-	2016-08-14	760.00																												




5	<p><b>* Quantity</b> <input type="text" value="6767"/></p> <p>The quantity field must be equal or less than the max quantity allowed.</p>	ok															
6	<p><b>* Book</b> <input type="text" value="978817596309"/></p> <p>The book is already added.</p>	ok															
7	<p><b>* Dealer</b> <input type="text" value="Vijithayapa Bookshop"/></p> <p>Vijithayapa Bookshop Sarasavi Bookshop Masterguide</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Contact</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td>Vijithayapa Bookshop</td> <td>23, kohuwala</td> <td>Current</td> </tr> <tr> <td>Sarasavi Bookshop</td> <td>34, Nugegoda</td> <td>Current</td> </tr> <tr> <td>ST Education</td> <td>colombo 12</td> <td>Past</td> </tr> <tr> <td>Masterguide</td> <td>Colombo 14</td> <td>Current</td> </tr> </tbody> </table>	Name	Contact	Status	Vijithayapa Bookshop	23, kohuwala	Current	Sarasavi Bookshop	34, Nugegoda	Current	ST Education	colombo 12	Past	Masterguide	Colombo 14	Current	ok
Name	Contact	Status															
Vijithayapa Bookshop	23, kohuwala	Current															
Sarasavi Bookshop	34, Nugegoda	Current															
ST Education	colombo 12	Past															
Masterguide	Colombo 14	Current															
8		ok															

Table E.2 Test results for the Sales Module

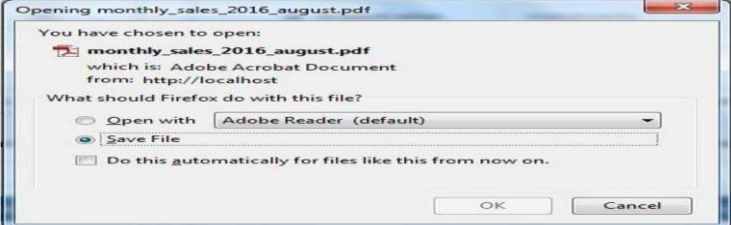
### Stock Module

Test No.	Actual Result [screen shots]	Status																				
1	<table border="1"> <thead> <tr> <th>ISBN</th> <th>Description</th> <th>List Price</th> <th>Min, Max Stock</th> <th>current St</th> </tr> </thead> <tbody> <tr> <td>9788175963092</td> <td>Essential English Grammar Raymond Murphy Cambridge University Press</td> <td>200.00</td> <td>20 100</td> <td>65</td> </tr> <tr> <td>9788185618517</td> <td>Intermediate English Grammar Raymond Murphy Cambridge University Press</td> <td>250.00</td> <td>20 100</td> <td>53</td> </tr> <tr> <td>9788175378092</td> <td>Edexcel Mathematics Grade 10 Dan Laemon Pearson Edu.</td> <td>4150.00</td> <td>20 100</td> <td>35</td> </tr> </tbody> </table>	ISBN	Description	List Price	Min, Max Stock	current St	9788175963092	Essential English Grammar Raymond Murphy Cambridge University Press	200.00	20 100	65	9788185618517	Intermediate English Grammar Raymond Murphy Cambridge University Press	250.00	20 100	53	9788175378092	Edexcel Mathematics Grade 10 Dan Laemon Pearson Edu.	4150.00	20 100	35	ok
ISBN	Description	List Price	Min, Max Stock	current St																		
9788175963092	Essential English Grammar Raymond Murphy Cambridge University Press	200.00	20 100	65																		
9788185618517	Intermediate English Grammar Raymond Murphy Cambridge University Press	250.00	20 100	53																		
9788175378092	Edexcel Mathematics Grade 10 Dan Laemon Pearson Edu.	4150.00	20 100	35																		
2	<table border="1"> <thead> <tr> <th>Min, Max Stock</th> <th>current Stock</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td>20 100</td> <td>65</td> <td>continue</td> </tr> <tr style="background-color: #f0f0f0;"> <td>20 100</td> <td>12</td> <td>continue</td> </tr> <tr> <td>20 100</td> <td>35</td> <td>continue</td> </tr> </tbody> </table>	Min, Max Stock	current Stock	Status	20 100	65	continue	20 100	12	continue	20 100	35	continue	ok								
Min, Max Stock	current Stock	Status																				
20 100	65	continue																				
20 100	12	continue																				
20 100	35	continue																				

3	<p style="text-align: center;">The New Book is added successfully.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%;">9788173674890</td> <td style="width: 30%;">Edexcel Science Grade 11 Sandra Maxwell Pearson Edu.</td> <td style="width: 10%;">6150.00</td> <td style="width: 10%;">20</td> <td style="width: 10%;">0</td> <td style="width: 15%;">continue</td> </tr> <tr> <td></td> <td></td> <td></td> <td>100</td> <td></td> <td></td> </tr> </table>	9788173674890	Edexcel Science Grade 11 Sandra Maxwell Pearson Edu.	6150.00	20	0	continue				100			ok
9788173674890	Edexcel Science Grade 11 Sandra Maxwell Pearson Edu.	6150.00	20	0	continue									
			100											
4	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 40%;">9788173674899</td> <td style="width: 60%;">Edexcel Science Grade 11 Sandra Maxwell Pearson Edu.</td> </tr> </table> <p>* ISBN <input style="width: 200px;" type="text" value="9788173674899"/></p> <p style="color: red; font-size: small;">The ISBN field must contain a unique value.</p>	9788173674899	Edexcel Science Grade 11 Sandra Maxwell Pearson Edu.	ok										
9788173674899	Edexcel Science Grade 11 Sandra Maxwell Pearson Edu.													
5	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 20%;">ISBN</th> <th style="width: 50%;">Description</th> <th style="width: 30%;">List Price</th> </tr> </thead> <tbody> <tr> <td>9788175963092</td> <td>Essential English Grammar Raymond Murphy Cambridge University Press</td> <td>150.00</td> </tr> </tbody> </table> <p style="text-align: center;">The Book is edited successfully.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 20%;">ISBN</th> <th style="width: 50%;">Description</th> <th style="width: 30%;">List Price</th> </tr> </thead> <tbody> <tr> <td>9788175963092</td> <td>Essential English Grammar Raymond Murphy Cambridge University Press</td> <td>200.00</td> </tr> </tbody> </table>	ISBN	Description	List Price	9788175963092	Essential English Grammar Raymond Murphy Cambridge University Press	150.00	ISBN	Description	List Price	9788175963092	Essential English Grammar Raymond Murphy Cambridge University Press	200.00	ok
ISBN	Description	List Price												
9788175963092	Essential English Grammar Raymond Murphy Cambridge University Press	150.00												
ISBN	Description	List Price												
9788175963092	Essential English Grammar Raymond Murphy Cambridge University Press	200.00												
7	<p style="text-align: center;">continue  </p> <p style="text-align: center;">The Book is discontinued successfully.</p> <p style="text-align: center;">Discontinue</p>	ok												
8	<p style="text-align: center;">continue  </p> <p style="text-align: center;">Discontinue</p>	ok												

Table E.3 Test results for the Stock Module

### Report Module

Test No.	Actual Result [screen shots]	Status
1		ok

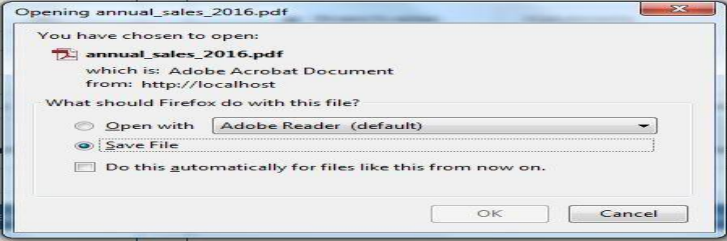
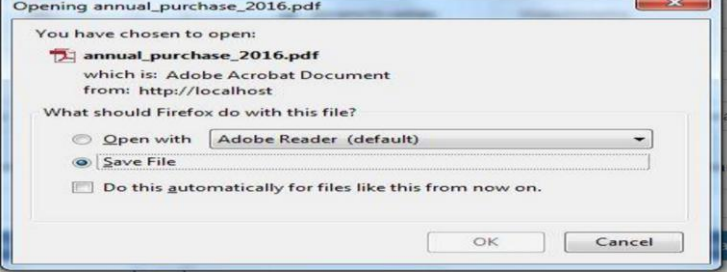
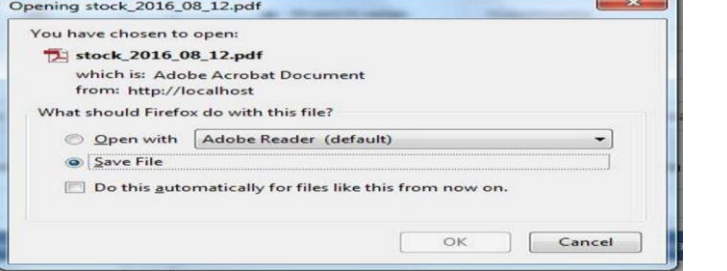
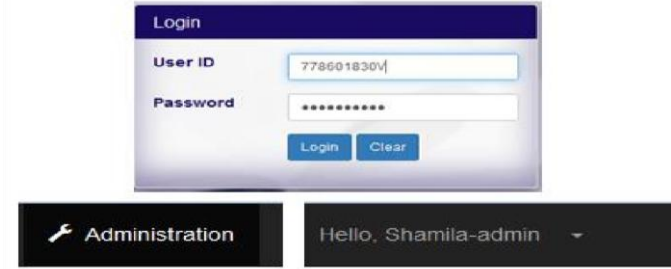
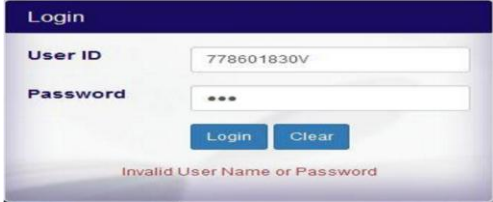

2		ok
4		ok
5		ok

Table E.4 Test results for the Report Module

### Administration Module

Test No.	Actual Result [screen shots]	Status
1		ok
2		ok
4		ok



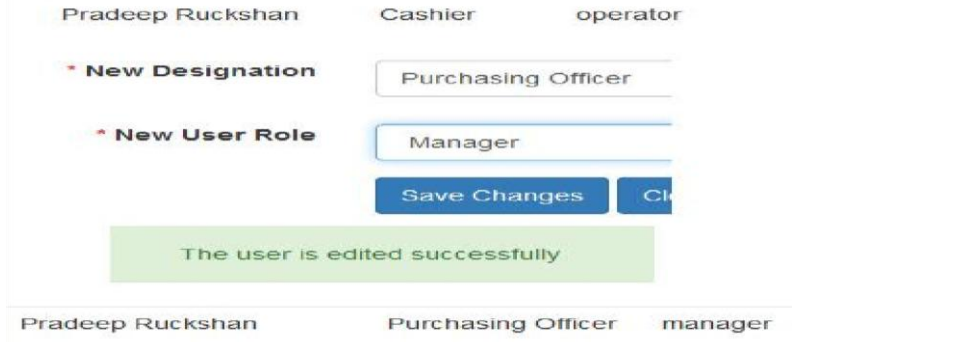
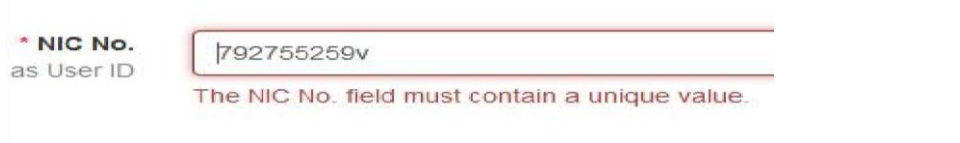
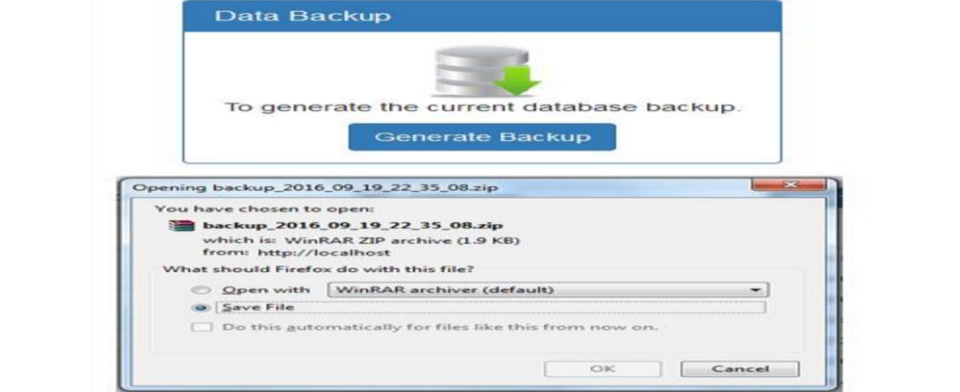
5		ok
7		ok
10		ok
12		Ok
14		ok

Table E.5 Test results for the Administration Module

## Assessment of User Feedbacks

As mentioned in evaluation chapter, structured questionnaire was used to capture the user feedback which is given in following table E.6.

Role manager	Poor	Average	Good	Excellent
<i>Interface</i>				
<b>Simplicity</b>			<b>x</b>	
<b>Understandability</b>				<b>x</b>
<b>Consistency</b>				<b>x</b>
<b>Easy to work with</b>				<b>x</b>
<b>Messages</b>			<b>x</b>	
<b>Help to avoid mistakes</b>			<b>x</b>	
<i>Process</i>				
<b>Close to the existing process</b>				<b>x</b>
<b>Improving the accuracy</b>				<b>x</b>
<b>Improving the efficiency</b>				<b>x</b>
<i>Reports</i>				
<b>Completeness</b>				<b>x</b>
<b>Relevance</b>				<b>x</b>
<b>Simplicity</b>				<b>x</b>

Table E.6 Sample questionnaire

## USER EVALUATION

In order to check the ideas of the end users regarding the system, a user evaluation feedback Form was prepared and distributed among all the employees and they filled the form and gave their personal views by going through the system. The following Figure E1 shows the user acceptance result sheet

**WISDOM**  
The Book World

**Wisdom Bookshop Web Based Bookstore -Wattala-**

**USER EVALUATION FEEDBACK FORM**

➤ User Name ..... A. Sutharshini .....

➤ User Designation ..... Manager .....

Please mark whether your satisfaction and Quality of the following Modules that are implemented in the Stock Control Management.

No	Modules	Poor / Fair / Good / Very Good / Satisfied
1	Purchasing Module	Very Good
2	Sales Module	Very Good
3	Stock Module	Very Good
4	Admin Module	Very Good
5	Report Module	Very Good
6	Maintenance of Product catalogue	Good
7	Maintenance of Stock catalogue	Very Good
8	Notification Sending	Satisfied
9	Help Tool	Good

➤ Remarks:

- \* It would be better to display detail views in pop up windows rather than load in separate tab.
- \* would be better to include some graphical repr of data to the reports.

Signature: *[Signature]* Date: 04/11/17

Figure E.1 User Acceptance Test Result Sheet

The Following E2 graph illustrated using the feedback given by each user through the above User Evaluation Form of WB.

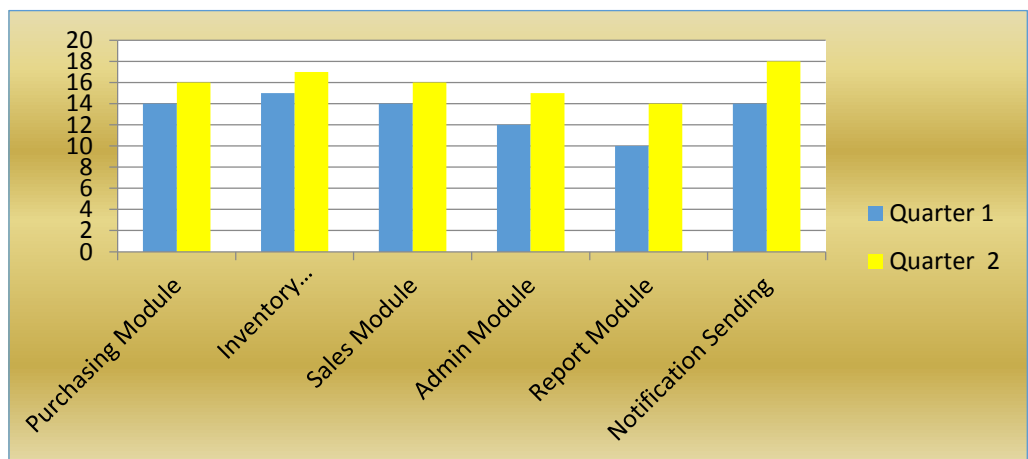


Figure E.2 User Evaluation Feedback chart in WB

Modules	Quarter 1	Quarter 2
<b>Purchasing Module</b>	14	16
<b>Stock Module</b>	15	17
<b>Sales Module</b>	14	16
<b>Admin Module</b>	12	15
<b>Report Module</b>	10	14
<b>Notification Sending</b>	14	18

Table E.7 User Evaluation Feedback Results

## Appendix F | Code Listings

This section provides some additional codes of the *Administration* controller, which is used for code explanation in the implementation chapter. Please refer the CD for complete codes.

The following code shows the *index()* function of the *Administration* controller, purpose of which is to load the view *user\_view* with data and the pagination.

```
public function index( $page = 0 ){
    $data          = $this->application->set_data();
    $data['module'] = $this->module;
    $criteria      = "";
    //set criteria if any
    if($this->session->userdata('users_is_search'))
        $key = $this->session->userdata('users_search_key');
        $value = $this->session->userdata('users_search_value');

        if($key == 'first_name' || $key == 'last_name') { //user name
    $key = "$key RLIKE";
        $value = "^$value";
    }else if($key == 'user_role') { //user role
        switch ($value){
            case '1': $value = 'operator' ; break;
            case '2': $value = 'manager' ; break;
            case '3': $value = 'admin' ; break;
            default: $value = FALSE;
        }
    }
}
```

The following code segment represents the function *view()* that is responsible for displaying the details of the specific *Administration*. The function retrieves the Administration details, and prepare them to an array *\$data* to send the view and then loads the view *user\_view* details.

```
<?php $this->load->view('templates/menu_temp'); ?>

<!--main area-->
<div class="row wbbm-application-main">
  <section class="col-sm-12">

    <!--tab navigation-->
    <br/>
    <section class="row">
      <div class="col-sm-12">
        <ul class="nav nav-tabs">
          <li class="active"><a>Users</a></li>
          <li><a href="<?php echo
site_url('administration/add'); ?>" >New User</a></li>
          <li><a href="<?php echo
site_url('administration/maintenance'); ?>" >Maintenance</a></li>
        </ul>
      </div>
    </section>

    <!--notification area-->
    <br/>
    <section class="row" id="notification">
      <?php
        if ($is_notification) {
          $this->load->view('templates/notification_temp');
        }
      ?>
    </section><!--end of notification area-->

    <!--search well-->
    <br/>
    <section class="row">
      <div class="col-sm-12">
        <section class="well well-sm text-right">
          <form class="form-inline" method="post"
action="<?php echo site_url('administration/user_search'); ?>"
onsubmit="return validate_search()" >
```



```

        <div class="form-group">
<span class="help-block" id="search_error"><p class="text-danger text-
left" id="search_error_text" ></p></span>
        </div>
        <div class="form-group">
            <select class="form-control input-sm"
name="lst_key" id="lst_key">
                <option value="nic" <?php if($this->session-
>userdata("users_search_key")==="nic"){echo "selected='selected'";}?>
>User ID</option>
                <option value="user_role" <?php if($this-
>session->userdata("users_search_key")==="user_role"){echo
"selected='selected'";}?> >User Role[1:operator, 2:manager
3:admin]</option>
                <option value="user_status" <?php if($this-
>session->userdata("users_search_key")==="user_status"){echo
"selected='selected'";}?> >User Status[1:active, 0:inactive]</option>
                <option value="first_name" <?php if($this-
>session->userdata("users_search_key")==="first_name"){echo
"selected='selected'";}?> >First Name</option>
                <option value="last_name" <?php if($this-
>session->userdata("users_search_key")==="last_name"){echo
"selected='selected'";}?> >Last Name</option>
            </select>
        </div>

```

The following code segment represents the function *deactivate\_user()* which is responsible for cancelling an existing users. It updates the user removes user account of the user as well. Finally, the confirmation message is generated.

```

public function deactivate_user($user_id){
    $criteria = array('nic' => $user_id);
    $data_set = array('user_status' => FALSE);
    $query_status = $this->user_model->update_user($data_set,$criteria);

```

```

if($query_status) {

    $description = "Deactivate user: {$user_id}.";

    $this->application->write_log('admin', $description);

    $notification = array(

        'is_notification'    => true,

        'notification_type'  => 'success',

        'notification_description' => 'The user is deactivated successfully.'

    );

}

else{

    $notification = array(

        'is_notification'    => true,

        'notification_type'  => 'error',

        'notification_description' => 'Error terminated deactivating the user.'

    );

}

    $this->session->set_flashdata($notification);
    redirect('administration');

```

As described before, it is very important to make server side data validation in order to ensure the accuracy of the data. Following code segment shows the *validate\_edit\_user()* function which makes the server side data validation when inserting the data of a new user.


```

function validate_edit_user() {
    btn_disable('#btn_edit_user');
    var status = true;
    var val ;
    var err ;
    clear_errors();

    val = $('#txt_email').val();
    err = $('#txt_email_error');

```

## Appendix G | Client Certificate

 **WISDOM**  
The Book World

**WISDOM BOOKSHOP**  
**'WISDOM BOOKS'**  
No. 310, Negombo Road,  
Wattala,  
Sri - Lanka.

Tel. & Fax No: +94 11 2948586 / +94 11 4897582  
Email: [wisdom@sltnet.lk](mailto:wisdom@sltnet.lk)  
Website: [www.wisdombooks.lk](http://www.wisdombooks.lk)

---

Date: 30<sup>th</sup> October 2017.

BIT Coordinator,  
University of School of Computing (UCSC),  
Colombo 7.

Dear Sir/Madam,

**Web Based Bookstore Management System for Wisdom Bookshop (WBBM)**

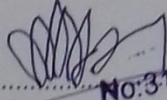
This is to certify that Miss Shamila Koralage(S.I.Koralage) has successfully Design and Developed a Web Based Bookstore Management System for Wisdom Bookshop (WBBM) in Wattala. The Project was undertaken by her as a partial fulfillment of a requirement for the Bachelor of Information Technology Degree.

The System has enabled the operations with Purchase, Sales, Stock, finance, Web and Admin of business activities. We inform this System will support us to carry business smoothly while providing a better Customer Service, Purchasing and Stock Control facilities connectivity of our daily business activities.

I believe that the system would be very helpful in achieving our business goals in the future.

Thank You.

Yours faithfully,

  
**WISDOM**  
No:310, Negombo Road,  
Wattala.  
Tel:2948586/0114897502

Alloy M. Robert.  
Managing Director,  
Wisdom Book Shop – Wattala.

# Glossary

**AJAX** – Stands for Asynchronous JavaScript and XML. Combine collection of technologies. Running on Client side and helps to develop interactive web applications.

**Apache** – Secure web server developed by Apache Software foundation.

**Agents** - person or organizations who supply the books

**Dealers** - person/ organization to whom the books sold

**WAMP** – is a software package of Open Source Bundled. Include Windows, Apache, MYSQL and PHP.

**Php** – Stand for PHP Hypertext Preprocessor. Object oriented supported server side scripting language.

**Interface** – Interconnect web system with the user.

**JavaScript** – Use to dynamic web applications. Develop by Netscape.

**JQuery** – is a popular java script library allows writing scripts with minimum coding. It is used extensively at present to integrate the dynamic effects into the web pages.

**Open Source** – means that the source code of the software is available for the users. Hence the users are able to modify the code if they need.

**Graphical User Interface (GUI)** – is a type of interface consists of various graphical elements such as buttons, menus and so on which is easy to use for the users.

**Structured Query Language (SQL)** – is the query language used to interact with the databases. One of most popular Database management system can handle big amount of data related to different types. Help to retrieve data base details.

**Database Management System (DBMS)** – is a type of software application facilitates the users to interact with the database.

**Joint Application Development (JAD)** – is a software development approach in which the end users are also involved continuously for the development process.

**RUP** – Stands for Rational Unified Process. Iterative software development methodology. Develop by Rational Software Co-operation.

**MD5 - (Message Digest Algorithm)** Use when converting data into unreadable format.

**UML** - Stands for Unified Modeling Language. Developed by Ivar Jacobson, James Rumbaugh, and Grady Booch at Rational Software. It is a modeling language.

# Index

## A

**Abstract** ..... iii  
**Acceptance Test**.....48, 89, 91  
**Activity diagram**..... 56, 60  
**AJAX** ..... iii, 33, 51  
**Analysis**..... 5, 14, 55  
**Administration**.....3, 24, 30, 49, 91

## B

**Backend** ..... 59  
**Backup**..... 10, 12, 23  
**Browser**..... 57  
**Button**..... 75

## C

**Client-server** ..... iii, 27  
**CodeIgniter** ..... 31, 34, 56  
**Controller** ..... 37, 40, 87  
**CSS** ..... iii, 33  
**Class Diagram**.....27, 66

## D

**Data set** ..... 31, 54,73  
**Database**..... 13, 28, 32, 68  
**Deployment**..... 65

## E

**ER Diagram**..... 23  
**Evaluation**..... 48, 89, 91

## F

**Framework**..... 35  
**Financial** ..... 11, 14, 60  
**Functional**..... 8, 10, 43

## G

**GUI**..... 12, 92

## H

**HTML**..... 33, 40, 55  
**http**..... 34, 38  
**Hardware**.....54

## I

**Integration**..... 31  
**Interface**..... 29, 56, 68  
**Interview**.....5  
**Invoice**..... 79

## J

**JavaScript**..... 32, 41  
**JQuery**..... 32, 42, 51  
**JAD** ..... 14, 92

## L

**local host**..... 55  
**Literature Review**.....10

## M

**Model**..... 35, 38  
**Modeling** ..... 12  
**MVC** ..... 28, 34  
**MySQL** ..... 32, 59  
**MD5**..... 38

## N

**Normal form**..... 25, 26  
**Normalization**..... 25  
**Notification**.....73

## O

**Object oriented** ..... iii, 15, 55  
**Observation**.....5

**P**

**PHP ..... iii, 32, 40**  
**Purchasing ..... 6, 10, 75**  
**Process Model..... 12**

**R**

**Report ..... 5, 23, 78**  
**Requirements ..... 5, 18, 57, 58**  
**RUP..... 12**  
**Reuse.....33**  
**RAD..... 12**

**S**

**Sales .....6, 8, 19**  
**Sequence diagram ..... 26, 60**  
**Server side ..... 94**  
**SQL ..... 39, 40**  
**Sampling.....5, 56**  
**Security..... 51**

**T**

**Test case..... 43, 45, 82**  
**Test data ..... 48**  
**Test plan ..... 42**

**U**

**UML ..... iii, 13, 51**  
**Unit testing..... 45, 54**  
**URL..... 34, 60**  
**Use-Case..... 18, 20, 57**

**V**

**Validation ..... 41, 94**  
**Verification..... 40**  
**View..... 22, 49, 77, 91**  
**Virtual admin..... 59**

**W**

**Wamp ..... 54, 92**  
**Web based system..... 15**